AN ARGUS SPECIALIST PUBLICATION

June/July 1984

uper
oftware prizes
o be won in our
ree to enter
competition

ZX COMPUTING

$3.95

£1.95

Britain's Best Magazine For The Sinclair User

Over 130 pages of information and programs for the ZX Spectrum, ZX81 and ZX80

ollow Slogo – Logo at a nail's pace!

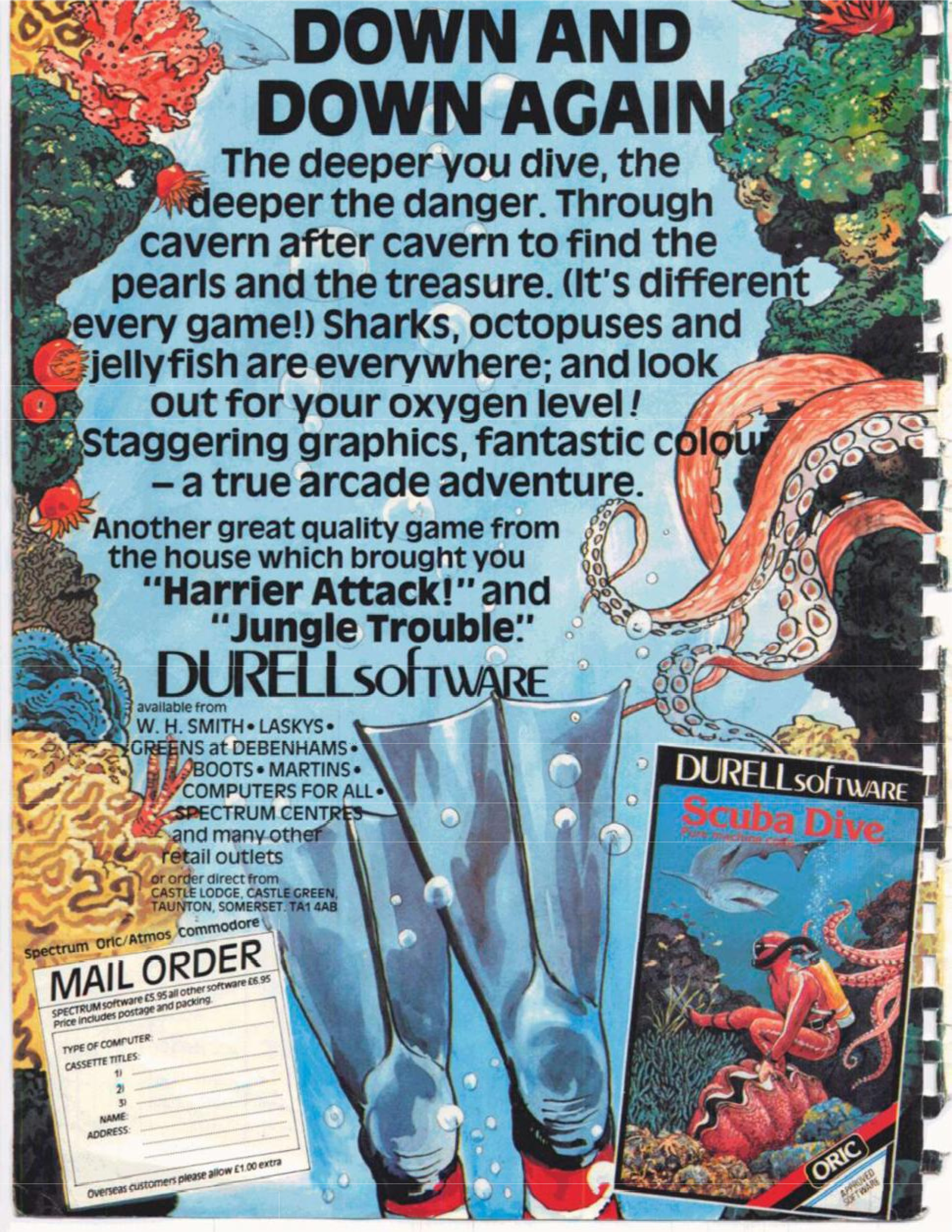


Education and domestic programs for your ZX81 and Spectrum

Get technical with our 3D graphics program

Learn BASIC the professional way

Going abroad? Cash in on our currency conversion



DOWN AND DOWN AGAIN
The deeper you dive, the deeper the danger. Through cavern after cavern to find the pearls and the treasure. (It's different every game!) Sharks, octopuses and jellyfish are everywhere; and look out for your oxygen level! Staggering graphics, fantastic colour – a true arcade adventure.
Another great quality game from the house which brought you "Harrier Attack!" and "Jungle Trouble."
DURELL software
available from
W. H. SMITH • LASKYS • GREENS at DEBENHAMS • BOOTS • MARTINS • COMPUTERS FOR ALL • SPECTRUM CENTRES and many other retail outlets
or order direct from
CASTLE LODGE, CASTLE GREEN, TAUNTON, SOMERSET. TA1 4AB
Spectrum Oric/Atmos Commodore
MAIL ORDER
SPECTRUM software £5.95 all other software £6.95
Price includes postage and packing.
TYPE OF COMPUTER:
CASSETTE TITLES:
1)
2)
3)
NAME:
ADDRESS:
Overseas customers please allow £1.00 extra
DURELL software
Scuba Dive
ORIC

LOOK!!

THE DATA RECORDER

DESIGNED FOR RELIABLE SAVING & LOADING OF COMPUTER DATA

- * COMPACT SIZE *(200mm x 130mm x 50mm)*
- * HIGH RECORD LEVEL
- * AUTO STOP
- * OUTPUT CUT ON RECORD *(leads left connected)*
- * AUDIO MONITOR ON LOAD & SAVE



BATTERY OR 6V ADAPTOR
ADAPTOR SUPPLIED

SPECTRUM £24.95

ZX81 £27.95
INCLUDING INTERFACE

GUARANTEED FULLY COMPATIBLE
WITH **SPECTRUM** & **ZX81** OR YOUR MONEY BACK!!

LOOK!! SUPER SOFTWARE NEW SPECTRUM TAPE PRICES!!

SPECTRUM 16K — PURRRFECT VALUE tapes for each computer — **ZX81 16K**

Road Race A 3D moving road is seen through the windscreen. You must stay on the road for 45 miles. Full control of gas, gears, steering. Full cockpit display.

Soccer 6 a side game for 2 players. Rebounding ball, moving men, score set, goals, fouls. Plan view of pitch.

Golf Play your way round a 9 hole random course. Each hole is set with bunkers, out of bounds & water hazards. 3D graphics. Full set of clubs, putting, full score display.

Bar Billiards 1 or 2 Players. Computer plays 1. Red ball, break and total score, plan view of table with mushrooms.

Kludo for 2 to 6 players with rooms, weapons, suspects. Full board display, random solutions and dealing. Good fun for all the family.

Battleships You battling the computer in this popular board game. Set your fleet, computers is random.

Stormforce An original board game for 2 players. Each player has 6 commandos and a field gun. Capture the enemies flag. Hidden mines and action graphics.

Starbuster You are hemmed in by a galaxy of stars, novas and super novas. Blast them to bits and move. High score chart.

Invasion Gunner Super fast MC! 3D moving display. Lasers, shields, 2 part invaders, 3 levels. Quick reactions needed!

Master of the Rings Avoid losing lives, Search snake pit, ravine, trees, maze, black rings, dark riders. All with traps Find 6 hidden rings. Action graphics.

SPECTRUM (£3.50 each) £6.00 for 2 tapes

2 SUPER GAMES ON EACH TAPE!!

ZX81 (3.50 each) £6.00 for 2 tapes

SPECTRUM 48K

NEW!!

FISHY DICK Dick must catch the fast moving fish as they swim in his river. He can cast and strike. Mind the ducks! Beautiful graphics All machine code. The more fish you catch the harder it gets! Cast & Strike with different bait for each fish.

£4.95

LOOK!!

PROGRAMMERS WORKSHOP

- ***Character Generator*** Block display generation of 21 UDG's + 147 extra in array storage.
- ***Memory Peeker***
- ***Tape Header Dissembler***
- ***Attributes Calculator***
- ***M.C. Sound Effects Generator*** 1000's of combinations. single note. ramped twin & two tone.
- ***Number Convertor*** Binary—decimal—hex up to 16 bit
- ***Number Translator*** *16 Bit To Lo & Hi order bytes in hex or decimal
- ***UDG Array Display*** Block displays of stored graphics

ONLY £4.95!

ALL Tape Prices include Postage and Packing and VAT
Data Recorder ADD £1.50 Postage, Packing & Insurance
Allow 7—14 days for delivery
Send cheques or Postal Orders only
Overseas orders for software tapes only! (ADD £1.00)

Meow Micros
8 Newnham Close
BRAINTREE, Essex CM7 7PR
Tel: (0376) 22425

ZX COM

Editor: Ray Elder
Editorial Assistant: Fiona Eldridge
Group Editor: Wendy J Palmer
Sales Executive: Penny Scoular
Advertisement Manager: Barry Bingham
Divisional Advertising Manager: Beverley McNeill
Copy Controller: Ann McDermott
Managing Editor: Ron Harris
Chief Executive: T J Connell

Origination and design by MM Design & Print, Circus House, 26 Little Portland Street, London W1N 5AF.

Published by Argus Specialist Publications Ltd, 1 Golden Square, London W1R 3AB.

CONTENTS

PUTING

ZX Computing is published bi-monthly on the fourth Friday of the month. Distributed by: Argus Press Sales & Distribution Ltd, 12-18 Paul Street, London EC2A 4JS. 01-247 8233. Printed in the UK by: Garnett Print, Rotherham and London.

Subscription rates: UK £12.70 including postage. Airmail and other rates upon application to ZX Computing, Subscriptions, Infonet Ltd, Times House, 179 The Marlowes, Hemel Hempstead, Herts HP1 1BB (Tel: 0442 48432)

ZX Computing is constantly on the look-out for well-written articles and programs. If you think that your efforts meet our standards, please feel free to submit your work to us for consideration for publication.

All submitted material should be typed if possible; handwritten work will be considered, but please use your neatest handwriting. Any programs submitted should be listed, a cassette of your program alone will not be considered. All programs must come complete with a full explanation of the operation and, where relevant, the structure; Spectrum programs should be accompanied with a cassette of the program as well as the listing.

All submissions will be acknowledged and the copyright in such works which will pass to Argus Specialist Publications Ltd will be paid for at competitive rates. All work for consideration should be sent to the Editor at our Golden Square address.

Welcome



Friday 1.15pm
Decided to come home to write introduction — too many distractions at the office.
Stare at blank wordprocessor screen.
"Hello, good evening and welcome."
No good, someone else has already used it.
Phone rings. Answer it, reminder from Fiona that introduction is due, return to blank screen.
Wife brings cup of coffee, put it on desk.
"Well, here we go again."
Corny, reach for coffee, spill it over notes.
Get cloth, return to find cat paddling coffee over desk.
Eject cat, mop up mess.

Friday 2.05pm
Screen still blank, like mind. Muse on emptiness of existence.
Youngest child enters clutching *Postman Pat* book, demands it be read to her. Eject child, child sobs, reproachful look from wife. Read *Postman Pat* to child followed by *Mr Men* and *Goldilocks and the Three Bears.*

Friday 2.55pm
Return to blank screen.
"What a great issue we have this time!"
Sounds too false, mustn't patronise readers.
Knock on door. Neighbour and friend, also computer nut, wonders if I'd like to see latest program? Possible source of inspiration! Accompany him to his abode. Spend time admiring space invaders program in BASIC, reluctantly agree that program too slow and not suitable for publication, yet another friend lost.

Friday 3.45pm
Return to own computer. Son has arrived home from school. "There was nothing on the screen so I loaded my program that I can't get working, can you help?"
Eject son; woeful looks from son, disapproving looks from wife.
Oh well, just a quick look.

Friday 4.25pm
Son departs to play on bike, phone rings. Fiona asking how the introduction is going, wonders if hysterical laughter is the sign that I've finally cracked. Return to Spectrum, daughter with friend of indistinct sex playing Hobbit. Decide to go for walk.
Return to fold after "just popping into the shop for a few things as you're going out."
Wonder if I've going out."
Wonder if I've got a hernia.

Friday 5.10pm
"Once again the indust. . ."
"Tea's ready, its getting cold, You've been in there all day."
"Yes dear."

Friday 5.55pm.
Cat returns to house proudly presenting us with a mouse. Mouse not dead. Mouse very much alive. Mouse escapes.
Children, cat and Editor try to encourage mouse from under fridge. Mouse not stupid. Wife refuses to wash up with a wild animal in the kitchen. Do washing up, anything for a peaceful life. Wife not stupid.

Friday 6.30pm.
Where was I? Why didn't I become a mountaineer or something else relatively sedate? Muse on alternative employement.

Friday 7.45pm.
Woken by children asking if I've fallen asleep, reply "just resting my eyes".
Try to prove it by typing "thr quijk bron foz" ego suffers from gales of youthful laughter.
Decide to watch TV for inspiration.

Friday 11.05pm
Les Dawson, The Friday Film and Match of the Day all fail to mention computers. Decide not to pay TV licence.
Back to the empty screen.

Friday 11.50pm
Screen still empty. Go to bed.

Unfortunately this month there is no introduction due to technical problems beyond my control.

A matter of form

Thank you all for the massive response to the questionnaire in the Feb/Mar issue. I had hoped to process them and print the results of all your opinions, but they are still arriving even as we go to press with this issue so, hopefully, all will be revealed in the next edition.

I have read many of them and my thanks to those readers who sent long detailed letters, I have read and digested them even though I may not have been able to reply personally.

One comment that a few readers have made is that *ZX Computing* is too patronising. This is certainly not intended; indeed, most contributors are of a standard where they could patronise me. My regard for ZXC's readers is high indeed, and one error brings many letters. The only thing I ask you take into account is that we have a large number of readers of different nationality, so I tend to write in as clear and simple a form as possible.

ZX81 revival!

I recently moaned about the lack of new ZX81 programs. I did mean commercial programs — the amount of readers' ZX81 submissions in the office is as great as for the Spectrum.

I'm therefore pleased to see that Softchoice Ltd of 52 Platts Lane, London NW3 7NT have entered the market with a range of ZX81 and Spectrum programs. The company is a subsidiary of a Canadian organisation and all their programs are well packaged and have been on sale in Canada and in the USA for some time. I wish them every success — anyone who supports the '81 deserves it! Several other producers of ZX81 tapes also sent samples along, my thanks to all of you and I'll try to keep our readers informed of your products either through Nick Pearce's reviews or the News columns.

Contributions

We are always on the lookout for good programs and articles for future issues of *ZX Computing* and where better to look than to our own readers. If, when reading through the magazine, you think you can write programs as well, or better than our present contributors, then let's hear from you.

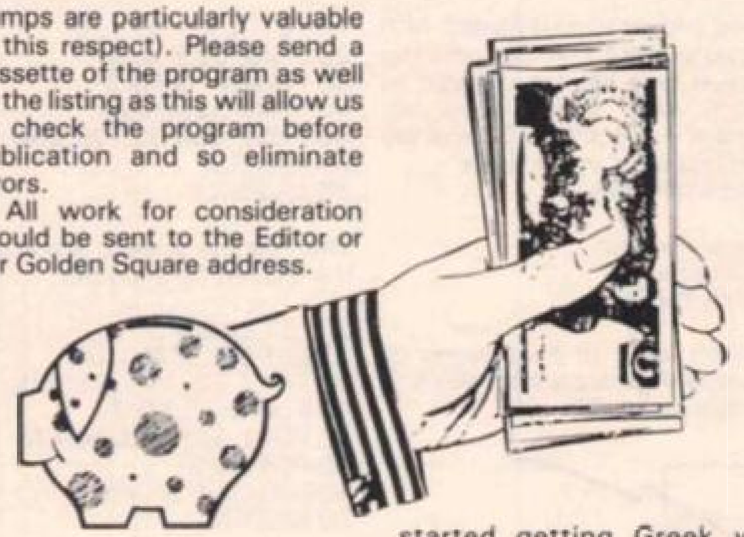
All contributions are paid for at very competitive rates, so if you've got your eye on a new ZX add-on or you'd just like to supplement your pocket money, then get writing! It is vital, though, that all the programs you send to us are totally original and not 'borrowed' or 'adapted' from othrr magazines or books. (When Tim Hartnell was Editor, he even received 'original' contributions he himself had written for his own books!)

The more original the program, the better chance it has of being accepted. Just another space invaders/duckshoot/frogger-type game, however good, is obviously not as appealing as a totally original idea. Any kind of program (business, domestic, educational or just fun) will be welcomed but particularly those which incorporate machine code or use ZX BASIC in clever or efficient ways or employ routines which can be re-used in other programs.

All submitted material should be typed if possible; handwritten work will be considered but please use your neatest handwriting. Program listings are vital, along with a clear explanation of how the program is constructed, what it does and what the user can expect to see when the program is RUN (Screen

dumps are particularly valuable in this respect). Please send a cassette of the program as well as the listing as this will allow us to check the program before publication and so eliminate errors.

All work for consideration should be sent to the Editor or our Golden Square address.



And now, the end is near...

(I'm not called 'Old blue eyes' for nothing!) I hope this issue of *ZX Computing* gives you hours of enjoyment and helps you to get the most from your ZX micro. Read on...

Ray Elder

A plea from the heart

Dear ZX Computing
I am twelve years old and in desperate need of a pen pal. I have a 48K Spectrum with a printer (what more could I want!) and would like to swap ideas with someone in my area. Can you help?
Yours faithfully

Jonathan Grimes
17 Little Strand Street
Skerries
Co. Dublin



Ingenious!

Dear ZX Computing
I recently purchased a ZX Printer. I was, at first, pleased with the print quality until I started getting Greek with French accents and UFO shapes among other things. I took it back to the shop and replaced it only to find that this one also gave me stretched and squashed characters.

There is a solution. I took a couple of small pieces of toilet tissue and fixed it onto the hubs which go into the roll of paper. The quality of the print is now super. There is, for those who believe otherwise, little need to even touch the paper while the printer is running.

I hope that I may have solved the problems of printer owners all over the country.
Yours faithfully

James McKay
Prestwick
Ayshire

Out for a duck?

Dear ZX Computing
As a relative newcomer to personal computing (though I have done a fair amount of programming on mainframes, and have even done such naughty things as play real-time, two-player Submarines on a time-shared line) I find your magazine very enjoyable and helpful. It is certainly more user-friendly than most other magazines I have seen, and I say this having only read your February issue in any depth (and no, I have no shares in Argus Press!)

ZX Computing certainly helps someone like myself, who feels the limitations of memory and speed of small computers very acutely after mainframes, get the most out of the Spectrum. The games you select, however long and tedious the keying-in may be, usually prove worthwhile and exciting.

I would, however, like to point out a few shortcomings in Duck Shoot (page 142). It contains some bugs and some inelegant points, that I think can be sorted out. To stop the gun falling off the screen and aborting the run (which it did in my implementtion) add the following lines:

```
72 IF e = -1 THEN LET e = 0
74 IF e = 27 THEN LET e = 26
```

Lines 20 and 107 seem unnecessary; also 96-99 are never called! Line 109 announces "out of bullets" too late, ie when the extra bullet is attempted. It is better to do this as soon as the last bullet has reached the ducks: do this immediately after checking for 10 hits, like this:

```
109 LET I=I+1
568 INK 2: IF I=135 THEN
    FLASH 1: PAUSE 30:
    PRINT AT 10,7: "OUT
    OF BULLETS!!!": BEEP
    3,-18: GO TO 600
```

The pause is to allow you to see whether you have hit that last duck!

There is a severe problem (as the program stands) with repeating the game (it seems to send you back to the last score). The following lines should do the trick:

```
3 LET x=0 (flag for replays)
5 LET a=0: LET c=10: LET
e=10: LET f=9: LET s=0: LET
I=0
21 IF x=1 THEN GOTO 1012
604 IF INKEY$ = "" THEN
GOTO 604
605 IF INKEY$ = "y" THEN IN-
VERSE 0: GOTO 5
610 CLS: INK 2 : PAPER 7:
PRINT AT 10,12; "END": Stop
```

Yours faithfully

John Kinory
London SW7

We occasionally use a 'less-than-perfect' program with the intention that readers will learn by modifying them themselves. We do try to make sure that they work though! Ed.



With all the trappings

Dear ZX Computing
I refer to your article, Trap It, by Mr D Pope on page 37 of your Feb/Mar 1984 issue.

In his machine code routine, it was necessary to load the DE register pair with the address of the start of the routine. This is achieved by the two instructions starting at label 'PUSH'. The startup routine dynamically modifies this first instruction and Mr Pope quite rightly points out that this is messy, but sees no alternative. There is, in fact, a 'clean' way to achieve the desired end.

The following simple piece of code can always be used to ascertain 'where we are' in storage. It hinges round the fact that a 'CALL' instruction pushes the absolute address of the instruction following it onto the stack.

```
CD5200 CALL82 ;CALL a RET
instruction that is located in ROM
3B ADDR DEC SP ;'Re-expose'
the last entry on the
3B        DEC SP :stack.
E1        POP HL ;Now HL will
contain the address of 'ADDR'.
```

Of course, the address of the RET instruction in ROM location 82 would probably be different for a ZX81, but any RET instruction will do.

Incidentally, if one merely wishes to disable the BREAK key, but do not require the sophistication of Mr Pope's routine, this may easily be achieved by inserting the following statement into the program:

```
LET anyvar = 23613: LET
anyvar = PEEK anyvar +
256 * PEEK (anyvar + 1):
POKE anyvar,0: POKE
(anyvar + 1),0
```

After this statement is executed, any report message (eg BREAK, STOP, O.K. etc.) will cause a system restart (equivalent to switching off and on). Obviously, one should only insert this statement when testing is complete!

I hope the above will be of interest to your readers, as well as Mr Pope.
Yours faithfully

Paul R Cohen
(pp Microcosm)
Ilford
Essex

INcredible

Dear ZX Computing
Back in the dark days at the beginning of December, I could joyfully have murdered anyone who mentioned the word 'IN' even if it wasn't computer connected. I'm much better now though — honest.

You see, at the end of November my Spectrum packed in and it was replaced. I had bought it in June and it must have been a series 2. The new one I now know to be a series 3.

Do you have any idea what effect it can have on you when a program which you know to be OK suddenly won't work? It was Earth Attack which appeared in your Oct/Nov edition and the control keys use 'IN' statements.

When I RAN it on the new Spectrum, the three Smart Bombs went off on their own, the lasers kept firing till they ran out, I got blown to bits and scored zero — I just sat there stunned because I hadn't touched a single key.

It took me four solid nights work of listing and re-listing the program until I finally narrowed it down to the main loop. At that time I knew nothing about 'IN' statements or how they worked but I realised that instead of waiting for input the computer was actually calling subroutines on its own.



The rest is history as they say and through trial and error I found that in series 3 Spectrums, the In Ports have a value of 191 when no key is pressed but on series 2 machines the number is 255. Now if I had not found this out I would have sent you a letter complaining about this and asking why it took so long for you to warn people in your magazine.

Instead may I offer you my thanks — I mean it — for in solving this problem on my own I have learned more in a couple of weeks about programming than I did in the 16 months after I bought my Spectrum.

I have enclosed a list of values for series 3 which should be read in conjunction with Chapter 23 of the Spectrum Manual. They might be of assistance to anyone still struggling with the problem.

Value	True When
191	No key pressed
191	Any key in ½ row pressed
190	1st key in ½ row pressed
189	2nd key in ½ row pressed
187	3rd key in ½ row pressed
183	4th key in ½ row pressed
175	5th key in ½ row pressed

Yours faithfully

Jim Thomson
Cumbernauld
Glasgow

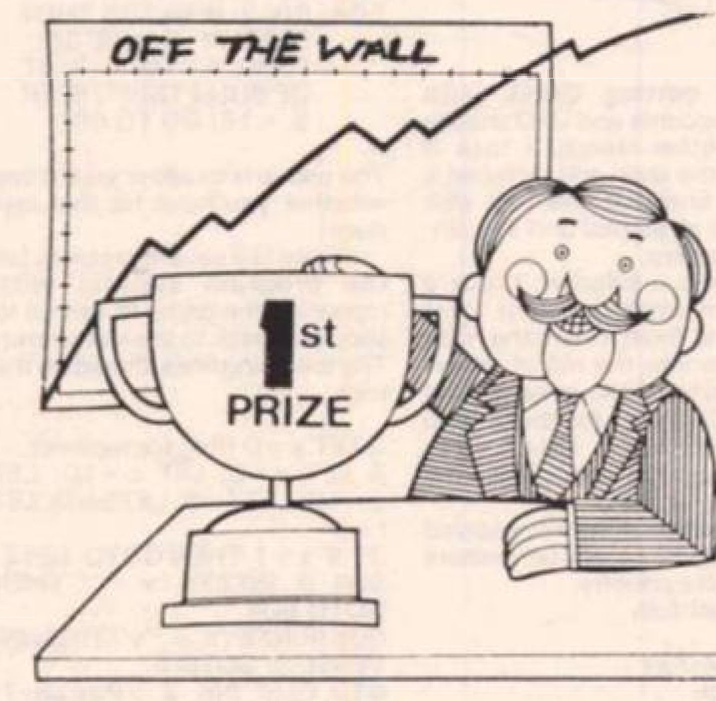
A new high?

Dear ZX Computing
After looking through the Dec/Jan issue, my brother decided to type in the Off the Wall program by Mr Elliston. Having finished, we set about trying to beat the top score of 513.

My brother broke this first with 517 after only getting 99 the first time but I beat him a few weeks later with 676. Nobody else has beaten this yet but I have improved on it myself with a score of 827 — this time the screen was almost covered to start with. Since then my brother has scored 638 and my dad has been in the 500s.
Yours faithfully

Jayne Pearson
Scarborough
North Yorks

I don't want to print pages of readers' high scores but this one looked good. Ed



Loads of code

Dear ZX Computing
As all Spectrum owners know the display file of the Spectrum is organised in a very complicated way. This makes PEEKing and POKEing to the display very difficult.

People who understand machine code will know that the only way to create fast flicker-free games is to POKE the display. This short machine code routine will calculate the address of any of the 5600 points on the display, therefore allowing users to PEEK and POKE anywhere on the screen.

To enter it, simply enter the LOADER program below and type in the numbers underneath. When you have finished, type NEW and save the program on tape using:

```
SAVE''code''CODE
32500,100
```

Here is the LOADER program and the numbers to type in:

```
10 CLEAR 32499
20 LET N=32500
30 INPUT I
40 POKE N,I
50 PRINT N,I
60 LET N=N+1
70 GO TO 30
```

```
1,0,0,120,230,248,31,31,
31,87,230,24,198,64,103,
122,230,7,15,15,15,129,
111,122,23,23,23,95,120,
147,254,0,40,4,36,61,32,
252,68,77,201
```

When you have finished type STOP. Now, enter PRINT USR 32500 and if all is well you should get the answer 16384, if not, or if the machine crashed, reload the program using:

```
LOAD'' ''CODE 32500,100
```

and check each byte of the program carefully. The way to use the program is as follows:

POKE 32502, LINE NUMBER

(Between 0 and 31)
POKE 32501, COLUMN NUMBER (Between 0 and 175)

If you now enter PRINT USR 32500, this will give you the address of the LINE/COLUMN coordinates that you POKEd into the 2 addresses. When POKEing in the LINE number, 0 is the top of the screen and 175 is the bottom.

The code can be stored as long as you change the LINE/COLUMN addresses appropriately.

Here are two programs illustrating exactly how the program works:

```
10 FOR N=0 TO 175
20 POKE 32502,N
30 FOR M=0 TO 31
40 POKE 32501,M
50 POKE USR 32500,255
60 NEXT M
70 NEXT N
```

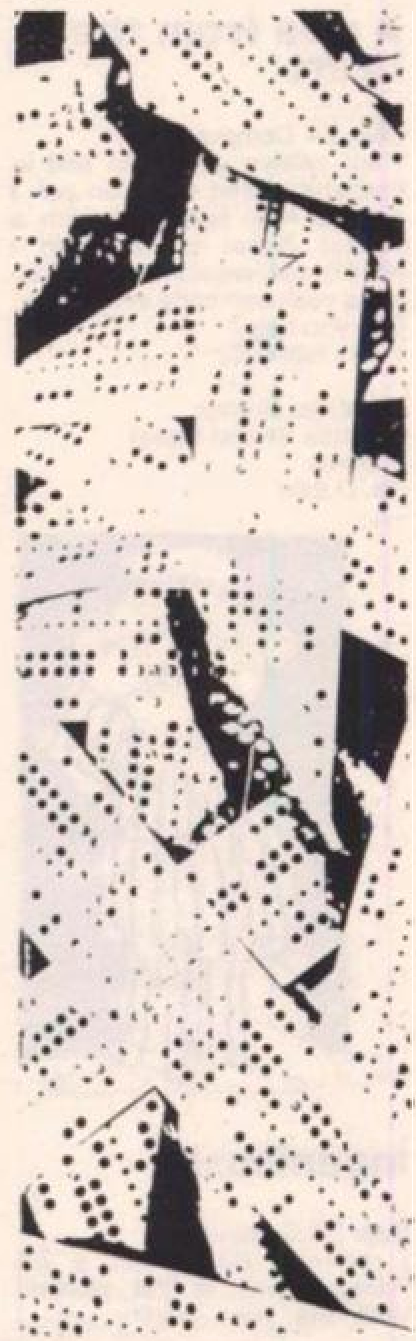
This fills the screen, starting at the top left hand corner.

```
10 LET LINE=0
20 LET COLUMN=0
30 POKE 32501,COLUMN
40 FOR N=0 TO 7
50 POKE 32502,LINE+N
60 POKE USR
32500,USR"a"+N
70 NEXT N
```

This prints the user-defined graphic A at position LINE/COLUMN on the screen.

I hope you can include this short article in a future issue of *ZX Computing* as I am sure it will be of use to all Spectrum owners.
Yours faithfully

Richard Wright
Richmond
Surrey.



The Editor strikes again

Galactoids, in the April/May edition of *ZX Computing*, was created as I explained by a program which created the DATA lines by PEEKing the machine code.

Unfortunately, over confidence got the better of me and I made a silly error by allowing for 16K of memory but forgetting that by CLEARing 30999 I had effectively created a 16K machine on a 48K. Needless to say, the computer won't let you do it and reports RAMtop no good.

However, the solution is simple; create the machine code higher in memory, NEW the DATA and then transfer it back down to its rightful place. To put it in simple stages, in Listing 2a, change:

```
2000 CLEAR 50999:
RESTORE
2010 FOR i=51000 TO
52417
STEP 10: LET c=0
```

and in line 2030, change 30990 to 50990

Now RUN the program (with all the DATA lines in of course).

Should you get an 'integer out of range' report, type PRINT i-50990 and press Enter. You will find that you have left a number of out of the DATA line printed on the screen.

Once the program has run successfully and reported O/OK then NEW it and Enter this line:

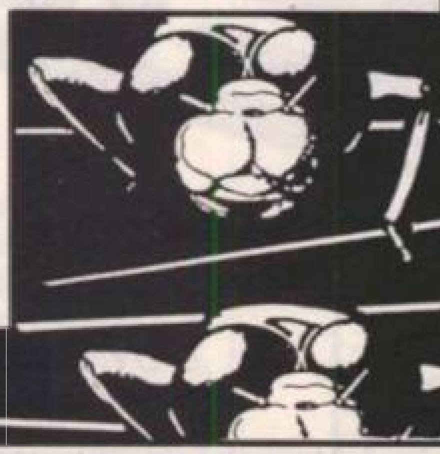
```
10 CLEAR 30999: FOR
i=31000 TO 32417: POKE i,
PEEK (i+20000): NEXT i
```

and RUN it. Finally, enter Listing 3 and save and continue as per the instructions printed.

My humble apologies to everyone who attempted Galactoids and especially to Gavin Smyth whose program I nearly ruined, and my thanks to Mr Redman for all his help.

Ray Elder







Errata

On page 117, ZXC April/May, Incentive Software were incorrectly given as the authors of Beyond Basic and Musicmaster for Sinclair. The company responsible for these packages is Incognito Software — our apologies for the mix-up.

The correct price for Gamestape 1 from Fawkes Computing is £5.95 and not £4.95 as printed.

THE FIRST IN DYNAMIC PACKAGES FROM PROCOM

WHAT IT IS

Whether you are a beginner or an experienced programmer it will make no difference with the DYNAMIC GRAPHICS package from Procom. You will be able to program professional looking graphics on your Spectrum. DYNAMIC GRAPHICS comes in two cassettes at the price you would normally pay for one. But do not let the low price put you off. DYNAMIC GRAPHICS is by far the most unique and much needed software aid on the market today.



£14·95
inc. VAT and P&P.

JUST LOOK AT WHAT WE HAVE IN STORE FOR YOU!

TAPE 1

THIS IS A CHARACTER DESIGNER WITH A DIFFERENCE * FULL SCREEN ANIMATION OF SPRITES * 27 FUNCTIONS INCLUDING SPRITE HANDLING 96 CHARACTERS WHICH CAN BE SAVED ON TAPE TO USE IN YOUR OWN PROGRAMS * EASY METHODS OF INCLUDING CHARACTER DESIGNS IN YOUR OWN BASIC PROGRAMS * NO KNOWLEDGE OF MACHINE CODE NEEDED * SCREEN SAVING * SPRITES MAY BE TREATED AS SUPER LARGE CHARACTERS (4 x 4). THE LIST IS ENDLESS. YOU WILL BE ABLE TO DESIGN LITERALLY THOUSANDS OF DIFFERENT CHARACTERS FROM PAC MEN TO SPACE MEN OR SPACE SHIPS TO BATTLESHIPS.

TAPE 2

THIS IS THE ULTIMATE UTILITY FOR COMPOSING HI-RES GRAPHICS ON YOUR SPECTRUM. AS WELL AS ESSENTIAL HAND DRAW CURSOR CONTROLS (OR JOYSTICK) THERE ARE MANY ADVANCED FEATURES, SUCH AS FAST FILL AND RESCALE. THERE ARE IN ALL OVER 50 COMMANDS. A CONSTANTLY UPDATED STATUS DISPLAY. GENERATED PROMPTS ARE FULLY DISPLAYED. NOW YOU CAN DESIGN SCREENS AS GOOD AS THE SOFTWARE COMPANIES.

1. DYNAMIC PROGRAMMING
2. DYNAMIC ANIMATION
3. DYNAMIC SOUNDS
4. DYNAMIC FILING
5. DYNAMIC EDITOR & ASSEMBLER

Package for the ZX Spectrum now!
(Also for the BBC, Electron and CBM 64)

EASY TO USE

DYNAMIC GRAPHICS is the one that everybody will understand. The instruction manual is simplicity itself. No more strained eyes — the manual has large 10 x 8 pages with plenty of diagrams to make it easy for you.

YOUR TICKET TO RICHES?

Use DYNAMIC GRAPHICS to make up your own program, and if it's good we will market it for you and maybe make you rich.

DYNAMIC GRAPHICS

HOW TO ORDER

Telephone orders (01) 508 1216

Credit Card Holders (Access/Barclaycard only) can purchase by telephone. Please give card no., name, address and items required, and your order will be despatched the same day.

BY POST

Simply write your requirements in the coupon provided, enclose your cheque/P.O or use your Access or Barclaycard. Please make cheque payable to; PROCOM SOFTWARE and post to address below. Allow 7 days for delivery.

IN PERSON

Visit our premises, at the address below for a great selection — we will be delighted to see you.

PROCOM SOFTWARE

309 HIGH ROAD
LOUGHTON
ESSEX
Tel: (01) 508 1216

Post to: PROCOM SOFTWARE, 309 HIGH ROAD, LOUGHTON, ESSEX.

Please send me the following items

- ☐ DYNAMIC GRAPHICS
- ☐ DYNAMIC PROGRAMMING
- ☐ DYNAMIC ANIMATION
- ☐ DYNAMIC SOUNDS
- ☐ DYNAMIC FILING
- ☐ DYNAMIC EDITOR & ASSEMBLER

(Please tick items)

I enclose my cheque/P.O. for £ ____________

I prefer to pay with my ACCESS/BARCLAYCARD
(Delete whichever not applicable)

Card No.
Signature
Name
Address

Tel: (Day)   Tel: (Eve)

Credit cards valid if signed by card holder.
Address above must be the same as card holder.

ZX4/84



Ack ack attack

Here's your chance to prove that you're a hero! Join the Dam Busters with C Stock of Chelmsford.

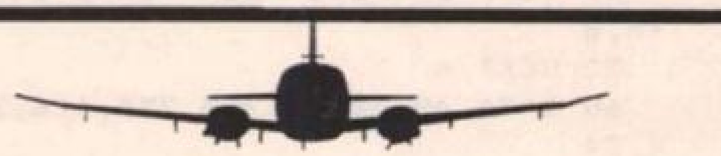
The enemy are on the offensive. Troops, guns, shells and tanks are pouring down into the enemy position along a single, heavily defended railway line. The only chance is for a single, daringly piloted mosquito plane to fly down the valley and destroy the railway bridge.

You have been selected and briefed for this task and victory or defeat rests with you.

The valley is randomly generated before the start of the game — this takes approximately 17 seconds but speeds up the running of the game. Four strings are defined, two for the valley and two for the guns.

Your plane appears at the top of the screen and the valley scrolls upwards toward you, as you go you must destroy the guns which are firing at you — 26 points — until you reach the bridge at the end of the valley.

When you drop your bombs the birds eye view changes to a side view of the valley and the trajectory of the bomb is shown followed by the results of your attempt. 100 points are awarded for a hit on the bridge, but beware, 100 points are lost if you miss the bridge, crash or are hit by the guns. If your score falls below −300 then the game is over — you failed to halt the attack and all is lost!



There are many variations on the "scrolling screen" game but type this one in and I'm sure you'll agree with me that this is one of the best.

Down the lines

48- 80	UDG s.
100- 180	Random valley.
1020-1195	Main loop of the program.
1085-1105	Checks if Ack Ack guns are in range, each gun has two shots at the plane.
1130-1190	Draw next line of the valley.
1200-1310	Ack Ack Guns.
1600-1670	Bombers cannon.
1800-1990	Bomb.
2001-2300	Finish.

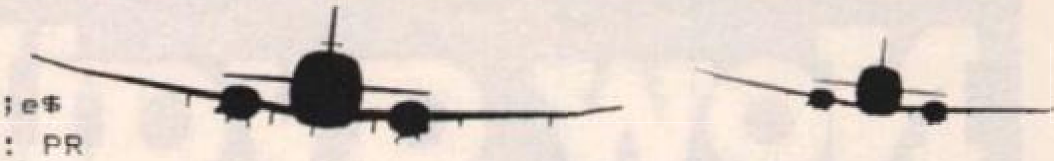




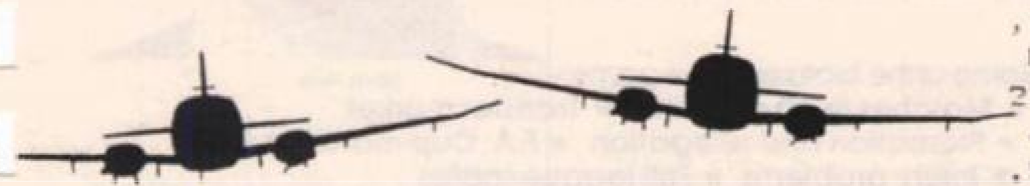




```
   2 REM -AA ATTACK- ? C.C.Stock
   3 REM Graphics A=A B=B C=C
D=D EF=EF
   4 PAPER 7: INK 0: BORDER 2
   5 PRINT  AT 5,4;"  ACK ACK  A
 T T A C K  "; AT 7,9; PAPER 6;"
 Please  wait "
  20 LET x=16: LET x1=16: LET a=
13: LET b=19: LET w=100: LET e$=
"
 "
  30 DIM a(100): DIM b(100): DIM
 c(121): DIM d(121): LET a(1)=13
: LET b(1)=19
  40 FOR n=0 TO 47
  45 READ q: POKE  USR  CHR$ 144
+n,q
  50 NEXT n
  60 DATA 60,24,24,255,255,126,2
4,24
  65 DATA 2,117,106,244,234,254,
126,56
  70 DATA 64,174,86,47,87,127,12
6,14
  75 DATA 255,129,129,165,255,36
,24,24
  80 DATA 0,0,240,248,255,255,25
5,0,0,0,0,120,254,255,254,0
  90 PRINT  AT 10,5;"BRIDGE =100
  CocB =26"; AT 12,5;"HIT/CRASH/
MISS = -100"; AT 16,2;"5=Left 6=
Bomb 7=Gun 8=Right"
 101 FOR n=2 TO 100
 120 LET a(n)=a+ INT ( RND *3-1)
: IF a(n)>25 OR a(n)<1 THEN  LET
 a(n)=a
 125 IF a(n)>a THEN  IF  RND <.3
4 THEN  LET c(n)=1
 130 LET b(n)=b+ INT ( RND *3-1)
-(b>30): IF b(n)-a(n)>8 THEN  LE
T b(n)=a(n)+8
 135 IF b(n)-a(n)<4 THEN  LET b(
n)=a(n)+4
 140 IF b(n)<b THEN  IF  RND <.3
4 THEN  LET d(n)=1
 180 LET a=a(n): LET b=b(n): NEX
T n
1001 REM ** ** ** valley
1005 PAPER 5: CLS : LET w=w-100:
 PRINT  AT 18,1;w
1008 PAPER 7: PRINT  AT 21,0;"XX
XXXXXXXXXX"; AT 21,19;"XXXXXXXX
XXXXX"
1010 IF w<-300 THEN  GO TO 2200
1020 FOR n=1 TO 100
1030 LET t=n-15
1035 PAPER 4: POKE 23692,-1: PRI
NT
1040 LET x=x-( INKEY$ ="5")+( IN
KEY$ ="8")
1045 IF  ATTR (5,x)=56 THEN  GO
TO 2300
1050 PAPER 5: PRINT  AT 5,x;"A";
 AT 4,x1;" ": BEEP .02,-8: LET x
1=x
1070 IF  INKEY$ ="7" THEN  GO TO
 1600
1080 IF  INKEY$ ="6" THEN  GO TO
 1800
1085 IF n<17 THEN  GO TO 1130
1095 IF c(t)=1 OR c(t-1)=1 THEN
 IF x-a(t-1)<4 OR x-a(t)<4 THEN
 GO SUB 1200
1105 IF d(t)=1 OR d(t-1)=1 THEN
 IF b(t)-x<4 OR b(t-1)-x<4 THEN
 GO SUB 1200
1130 IF n<100 THEN  LET a=a(n):
LET b=b(n)
1145 PAPER 7: IF c(n)=1 THEN  PR
INT  AT 21,a-1;"oB": GO TO 1155
1150 PRINT  AT 21,a;"X"
1155 IF d(n)=1 THEN  PRINT  AT 2
1,b;"Co": GO TO 1170
1160 PRINT  AT 21,b;"X"
1170 PAPER 5: PRINT  AT 21,a+1;e
$(2 TO b-a)
1172 IF n=100 THEN  PRINT  AT 21
,0; PAPER 6;"###################
#############"
1185 IF n>120 THEN  GO TO 1000
1190 IF n>99 THEN  LET n=n+1: GO
 TO 1030
1195 NEXT n
1201 REM ** ** ** AA Gun
1210 BEEP .02,4: LET x2=x+ INT (
 RND *3-1): LET y2= INT (4+ RND
*3)
1220 PRINT  AT y2,x2; PAPER 8;"*
": IF y2=5 AND x2=x THEN  GO TO
1301
1230 BEEP .02,4: RETURN
1301 PRINT  AT 5,x; PAPER 2; INK
 6; FLASH 1;"*"
1310 FOR n=1 TO 12: BEEP .1,( RN
D *8): NEXT n: GO TO 1000
1601 REM ** ** ** gun
1610 FOR g=6 TO 8
1630 BEEP .01,24: IF  SCREEN$ (g
,x)="X" THEN  GO TO 1080
1640 BEEP .01,24: IF  SCREEN$ (g
,x-1)="o" THEN  LET w=w+26: LET
c(n+g-21)=0: PRINT  AT g,x;"X":
GO TO 1080
1650 IF  SCREEN$ (g,x+1)="o" THE
N  LET w=w+26: LET d(n+g-21)=0:
PRINT  AT g,x;"X": GO TO 1080
1670 PRINT  AT g,x;"!": NEXT g:
GO TO 1080
```



```
1801 REM ** ** ** Bomb
1810 CLS : LET s=(122-n)*2
1820 PAPER 4: PRINT  AT 12,0;e$
1830 FOR n=13 TO 20: PAPER 7: PR
INT e$: NEXT n: PRINT ; PAPER 6;
e$
1850 IF s<0 OR s>31 THEN  GO TO
1900
1860 FOR n=12 TO 20: PRINT  AT n
,s; PAPER 5;"  ": NEXT n
1880 PRINT  AT 21,s;"DD"
1901 PAPER 7: PAUSE 50
1905 FOR n=1 TO 30
1910 PRINT  AT 15,n-1; PAPER 8;"
 EF"
1920 IF n>21 OR n<11 THEN  GO TO
 1980
1930 LET a=(n-11)^2/2.3
1940 PLOT n*8,47-a: DRAW 4,0: PL
OT n*8-3,45-a: DRAW 4,0
1945 IF n=21 THEN  PRINT  AT 21,
20; PAPER 2; INK 6; FLASH 1;"**"
```

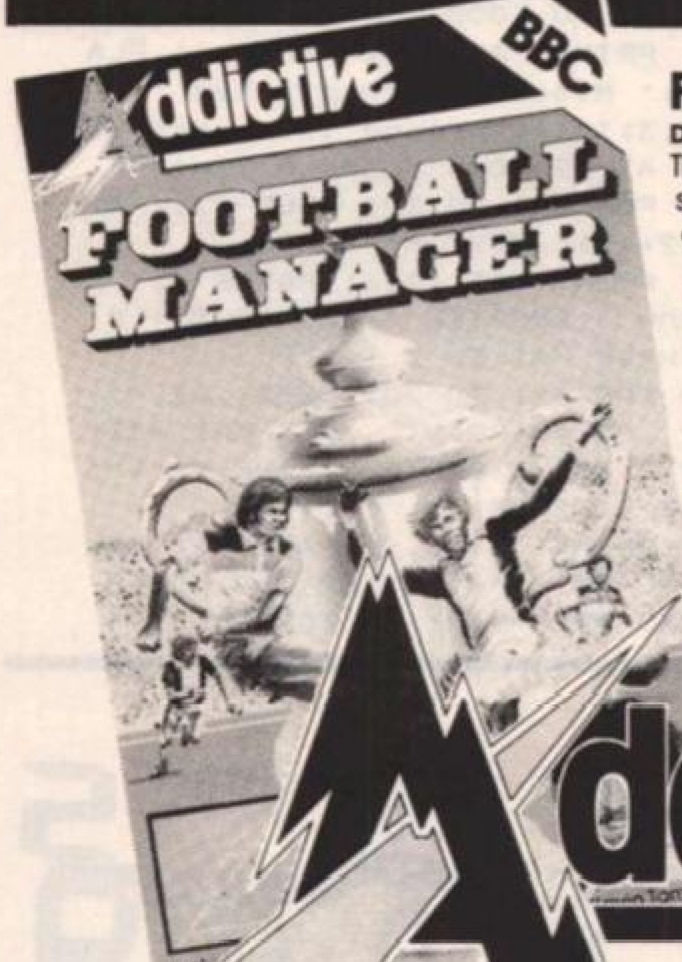


```
: BEEP 1,0: BEEP 1,0: BEEP 1,12
1947 IF n=21 AND s=20 THEN  LET
w=w+100: GO TO 2000
1950 NEXT n
1980 PAUSE 7: NEXT n
1990 GO TO 1000
2001 PRINT  AT 8,5; PAPER 6; FLA
SH 1;" Mission Accomplished "; A
T 10,3;" The bridge is destroyed
 "; AT 12,10;" Score = ";w;" "
2010 FOR n=1 TO 20: BEEP .2,( RN
D *12+12): NEXT n: STOP
2201 PRINT  AT 10,6; PAPER 4;" M
ISSION  ABORTED "; AT 12,5;" C
YOU HAVE FAILED B "
2210 FOR n=1 TO 30: BEEP .02,4:
BEEP .02,4: PRINT  AT ( RND *21)
,( RND *31); PAPER 6; INK 2;"*":
 NEXT n: STOP
2301 PRINT  AT 5,x; PAPER 2; INK
 6; FLASH 1;"*": BEEP 1,4: BEEP
.5,12: BEEP 1,0: GO TO 1000
```

The No.1 Football Game*

Now available for the BBC model 'B'

BBC Model 'B', ZX81 16K, ZX Spectrum 48K



Football Manager

Designed by Kevin Toms

This outstanding game of skill and strategy is now widely available and you can join the many thousands of satisfied customers who have purchased the game. This is no five minute wonder – you will be playing this game for hours over many weeks (we know – our customers tell us!).



Kevin Toms

Some of the features of the game:–

- ★ Matches in 3D graphics ★ Transfer market
- ★ Promotion and relegation ★ F.A. Cup matches
- ★ Injury problems ★ Full league tables
- ★ Four Divisions ★ Pick your own team for each match
- ★ As many seasons as you like
- ★ Managerial rating
- ★ 7 skill levels
- ★ Save game facility

Addictive

* ZX81 Chart
Home Computing Weekly 21.2.84

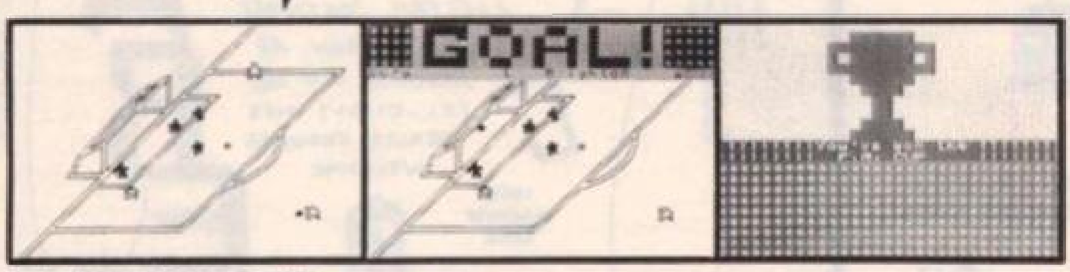
Programmers
We want your software. Send us a sample, if accepted, we will supply generous royalties plus cash advances plus free advice from Kevin himself!

Comments about the game from press and our customers

"FOOTBALL MANAGER is the best game I have yet seen on the Spectrum and my personal favourite of all the games on any micro ... To the ordinary person it is an excellent view of what can be done in the field of computer games ... The crowning glory of this game is the short set pieces of match highlights which show little stick men running around a pitch, shooting, defending and scoring ... It is a compulsive game but people who cannot take game sessions of 9 hours or so, which happened on one happy Sunday, will be grateful to know that there is a 'save to tape' option. FOOTBALL MANAGER has everything it could ... The originator, Addictive Games, certainly deserve the name." Rating: 19/20 (Practical Computing – August 1983).

"When I first received this game I spent the best part of the weekend playing it. Since then I have returned to it more often than any other. The truly addictive quality of this game is the remarkable way it mirrors the real football manager's problems ... "(Personal Computer Games – Summer 1983)

Versions coming soon for
Commodore 64
Dragon, Oric
and Atmos.



Action from the Spectrum version

"I am writing to say what a great game it is. I have spent over 45 hours on it." (Mr. D. Fearn – Gloucestershire.)

Available from computer software stockists nationwide, including Boots & John Menzies

Prices:
BBC Model B £7·95
Spectrum 48K £6·95
ZX81 16K £5·95
(N.B. 3D GRAPHICS ARE NOT INCLUDED IN THE ZX81 VERSION).
Overseas orders add £1.50

To order by mail (p&p free) send cheques or postal orders to:

Addictive Games
7A RICHMOND HILL, BOURNEMOUTH BH2 6HE

Dealers! For urgent stocks send your headed notepaper direct to our address.

Gamesmanship

Be one up on your friends with some excellent advice from Tim Hartnell on how to improve your programming.

It is very easy to generate startling graphic displays on the Spectrum as you well know. A few natty user-defined graphics, a flash or two on the border, and you're well on the way to producing a great game.

However, the very ease of producing superb displays on the Spectrum can be quite a trap on the road to producing worthwhile programs. If you're like me, at one time or another you've probably seen a great game at an arcade and immediately rushed home with the idea of writing the same game (or something similar) on your Spectrum. You have a pretty good idea of what the screen looks like when the game is underway, and you immediately start to re-create the screen display.

This is the trap. While it is fairly easy, given a little patience, to recreate a arcade-game screen fairly faithfully on the Spectrum, even when working in BASIC, it is far from easy to then produce a game which is worth playing. I suggest that you may consider working, from now on, the other way around.

If a game is to be any good on a computer, it must fulfil a number of criteria. A superb screen display is of little use if the game behind it is so poor that it rapidly becomes boring. Far better a game which just uses text output, or chunky graphics, rather than complex user-defined graphics, if the text program is actually worth playing. Far better that you write an intriguing game with a merely adequate display, than write one which looks good but which has little intrinsic merit. May I suggest that you try writing the game first, without concentrating at all on the graphics (simply using a non-defined graphic 'A' and so on) until you get a game which is fun to play, even without colour, BEEPs or user-defined graphics. Then, when you add these, a good game will be made even better. And, more to the point, all your energy when you are fresh and starting a program, will have gone into making the game worthwhile, rather than make your aliens look more ferocious. Get your program working first, and then dress it up.

The important points

Towards the end of this article, I'll give you a number of short routines which you can use for dressing up programs. However, first I would like to expand on my earlier statement that there were a number of criteria which determined whether or not a computer game was going to be any good.

You can probably add a few to the ones I'm going to outline from your own experience. Think about the arcade games you most enjoy playing. What makes them so enjoyable? It is highly likely that others will enjoy games for the very same reason you do, so if you can analyse the reasons for one game's success, you may well be able to use that knowledge to improve your own programs.

The objective of the game must be clear to the player. It might be to blast away all the weeds on the screen with your can of weedicide, or to find the Holy Grail, or to move from the left to the right of the screen without being blown up or whatever. The purpose of the game should be clear fairly quickly to the player and, in general terms, a means to that goal should also be clear. One thing that held the Tron arcade game back in its early days was that it was pretty difficult to work out what to do and, even when you had, to many players it hardly seemed worth the trouble. So, an early step in game development is to work out exactly what the player is supposed to do to win the game.

The path to be followed to that goal must also be reasonably evident. Although the purpose of the game should be clear, it is much more important that once a player starts playing a game, he or she should be developing skill at it. I cannot stress how important this aspect is. A game should be sufficiently hard to ensure that it cannot be mastered the first time it is played, but should not be so difficult that the player gets no positive feedback at all. However, the player should be developing games skills while playing, so that the next time he or she approaches it, a better score is likely. Would you tackle Pole Position over and over again if you didn't even qualify after fifty attempts at the game?

To ensure that your game includes factors which the player can become skillful at manipulating, you need to minimise the random element of the game. I do not mean that you should *not* have random factors in a game, to ensure that each game is different from the preceding ones, but that the randomness should be used carefully.

For example, the flight of a mother ship across the top of the screen could be triggered randomly, but the effect of pressing the 'F' (for fire) key should not be random. How could you master Ant Attack if you could never be sure how far you could lob a grenade?

So, the first element in developing a game with which the player can develop playing-skill is careful use of randomness. In adventure programs this is particularly important. The map you develop as you work your way across the Doomed Land should relate to a stable environment or the Adventure degenerates into a 'game' of randomly-triggered PRINT statements. Aliens which die only some of the time, or a 'rotate' command which only swings your craft around every so often is a certain way to produce a game which will not encourage repeated plays.

The rewards of playing should be clear, and worth the effort. It is unfortunate that the widely (and justly) acclaimed Microsoft Flight Simulator (plus several not-so-famous ones, including few for the Spectrum) uses up so much memory controlling the game that there is nothing left at the end except the pathetic word CRASH in the middle of the screen (or YOU HAVE LANDED SAFELY, if you're skillful). Let the screen break out in technicolour rainbows and make the most of the feeble little BEEP when someone wins a game, so their effort will be worthwhile.

Added to this is the use of high scores. These are very easy to include and are perhaps the simplest way to maintain interest as a player goes back time and time again to try and beat his or her previous best effort.

This way up

I have repeatedly advocated that games should be written in a top-down manner, calling the whole program from a series of subroutines which lie within a loop. Part of this 'structured' approach is to have the initialisation material at the end of the program, so that the start of the game looks like this:

```
10 REM WHIZ-BANG ALIENS
20 GOSUB 9000:REM INITIALISE
30 GOSUB 1000:REM PRINT SCREEN
40 GOSUB 2000:REM MOVE ALIENS
50 GOSUB 3000:REM PLAYER RESPONDS
60 GOTO 30
```

In the section from line 9000, you include the line LET SCORE=0. To include a high score feature, you add before line 20 the following:

```
15 HISCORE=0
```

Now, wherever you are within the program when it ends, you include the following (the line numbers are, of course, arbitrary):

```
500 IF SCORE > HISCORE THEN LET HISCORE=SCORE: PRINT "THE NEW HIGH SCORE IS ";HISCORE
510 PAUSE 500
520 GOTO20
```

As you can see, the GO TO 20 sends action to the subroutine call for the initialisation routine

again, but does not hit line 15, which originally set the high score to zero. This means that the whole program begins from scratch, except the value of the high score. (By the way, if you're running this on a non-ZX computer, you may have problems in redimensioning arrays so they may well have to be before line 20 as well.)

Finally in this article, I'd like to share with you a number of routines developed by David Perry of Donegore, Northern Ireland, which can you can add into functioning programs to make them more attractive, useful of efficient.

The first one (Fig 1) is a circle routine, which uses the sine/cos equation at an amazing speed to draw a circle in much the same way as the Spectrum's circle command does. X is the X co-ordinate, Y the Y co-ordinate and R the radius:

```
   1 REM    DRAWING CIRCLES
   2 REM  (C) DAVID PERRY 1984
  10 LET x=120: LET y=85: LET r=
50
  20 LET oa=x+r*COS (0): LET ob=
y+r*SIN (0)
  30 PLOT oa,ob
  40 FOR n=.3 TO 6.39 STEP .3
  50 LET a=x+r*COS (n): LET b=y+
r*SIN (n)
  60 DRAW a-oa,b-ob: LET oa=a: L
ET ob=b: NEXT n
```
(Fig 1)

The next one (Fig 2) allows you to paint in the contents of a circle within a BASIC program. The variables are as in the first program.

```
   1 REM        CIRCLE FILL
   2 REM  (C) DAVID PERRY 1983
   3 CLS
  10 INPUT "X-CORDINATE (0-255)
";X
  20 INPUT "Y-CORDINATE (0-175)
";Y
  30 INPUT "RADIUS ";R
  35 CIRCLE X,Y,R
  40 FOR N=0 TO R
  50 LET Z=R↑2: LET ZZ=N↑2
  55 LET XX=SQR (Z-ZZ)
  60 LET YY=Y+XX: LET XY=2*XX
  70 PLOT X+N,YY: DRAW 0,-XY
  80 PLOT X-N,YY: DRAW 0,-XY
  90 NEXT N
 100 PRINT AT 21,0;"PRESS ANY KE
Y": PAUSE 0: RUN
```
(Fig 2)

Our next program (Fig 3) allows you to rotate a user-defined character clockwise. This means, for example, you would only need to define one ship for a game like Asteroids and then get the routine to rotate it.

```
   1 REM    ROTATE CHARACTER
   2 REM  (C) DAVID PERRY 1984
   5 CLS : INPUT "CHARACTER TO B
E ROTATED (A-U)"; LINE A$
   6 IF A$<"a" OR A$>"u" THEN GO
 TO 5
  10 LET A=USR A$: LET C=A
  20 LET Q=1: LET W=8
  40 DIM Z(8,8): FOR D=1 TO 8
  50 LET X=128: LET B=PEEK A
  60 FOR F=1 TO 8
  70 IF (B-X)>=0 THEN LET B=B-X:
 LET Z(Q,W)=1
  80 LET X=X/2: LET Q=Q+1
  90 NEXT F: LET Q=1: LET W=W-1:
 LET A=A+1: NEXT D
 100 FOR X=1 TO 8: LET A=0: LET
B=128
 110 FOR Y=1 TO 8
 120 IF Z(X,Y)=0 THEN PRINT " ";
: GO TO 140
 130 LET A=A+B: PRINT "█";
 140 LET B=B/2: NEXT Y: POKE C,A
: LET C=C+1: PRINT : NEXT X
 150 PRINT : PRINT "LOOK AT THE
NEW GRAPHIC '";a$;"'!"
```
(Fig 3)

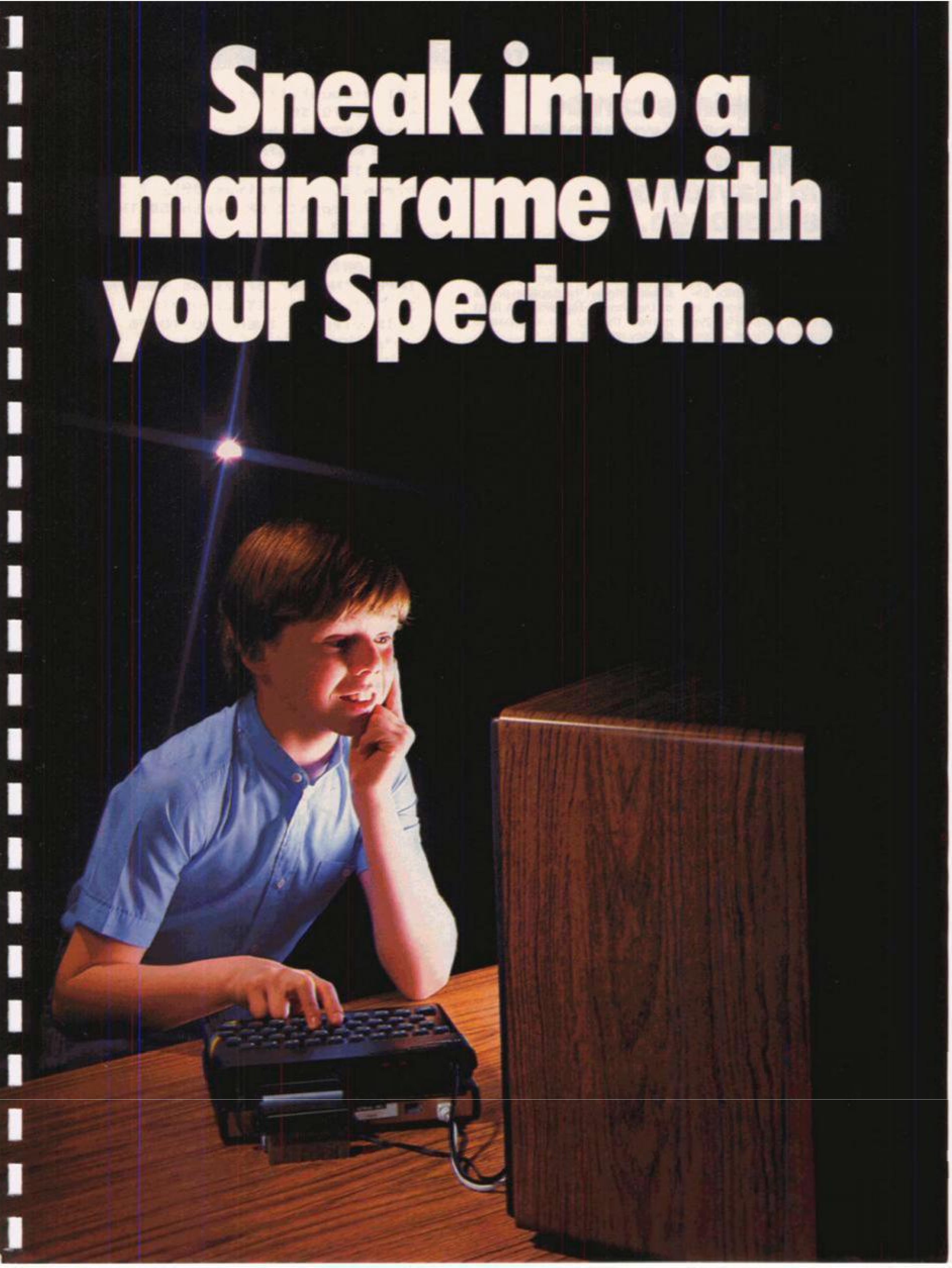
This program, (Fig 4) which shows how to scroll any string of more than 32 characters, would make a very impressive 'YOU HAVE WON' frame, or a good start to a game. You'd probably have to slow it down a bit. As you'll see, it happens very, very quickly, so it is difficult to read the demonstration program.

```
   1 REM SCROLLING A STRING
   2 REM  (C) DAVID PERRY 1983
   3 CLS
  10 LET A$="
               THIS IS AN EXAMPLE
PROGRAMME TO SHOW HOW FAST THE Z
X-SPECTRUM CAN ACTUALLY SCROLL A
 STRING ACROSS THE SCREEN.  IT I
S SO FAST THAT YOUR BRAIN GETS M
UDDLED UP UNTIL YOUR EYES BEGIN
TO HURT.              IF YOU LOOK
 AT THE PROGRAMME TO DO THIS YOU
 WILL FIND THAT YOU CAN HAVE ANY
 STRING OF CHARACTERS IN A$ AS L
ONG AS THERE IS MORE THAN 32!. H
APPY PROGRAMMING!!!!!!!!!!!!!!

 "
  20 LET X=1: LET Z=LEN (A$)
  30 PRINT AT 0,0;A$(X TO X+31)
  40 LET X=X+1: IF (X+31)=Z THEN
 STOP
  50 GO TO 30
```
(Fig 4)

Finally, here's a program (Fig 5) which accepts a string of characters and then prints them out sideways on the printer, either eight or sixteen times larger than normal. The people who make printer paper love programs like these!

```
   1 REM        BANNER PRINTER
   5 REM  (C) DAVID PERRY 1984
   6 REM
  10 INPUT "WORD (MAX 32 CHARACT
ERS)!",A$
  11 LET L=LEN (A$): IF L>32 THE
N PRINT "SORRY TOO LONG": GO TO
10
  15 INPUT "SIZE (1 OR 2)";S
  16 IF S<1 OR S>2 THEN GO TO 15
  17 LET T=1: IF S=2 THEN LET T=
.5
  30 INPUT "0:NORMAL & 1:INVERSE
";I
  36 IF I<0 OR I>1 THEN GO TO 30
  37 LET A=2: LET B=0: PRINT AT
0,0; INVERSE I;A$
  40 FOR X=0 TO (L*8)-1 STEP T
  50 FOR Y=166 TO 175 STEP T
  60 IF POINT (X,Y)=1 THEN LPRIN
T AT A,B;"█";
  70 LET B=B+1: NEXT Y: LET B=0:
 LPRINT : LET A=A+1: NEXT X
  80 CLS : GO TO 10
```
(Fig 5)


Sneak into a mainframe with your Spectrum...

Craft drawings can be viewed in several ways with this program from Martin Clayton of Knaresborough.

When you have entered this program and first run it, a brief explanation of how it works is given. To get the most effective displays from it however, a few moments of preparation is necessary.

The shape must be a prism although the cross section can have any number of points. Curves are not allowed — even VU-3D, one of the best drawing programs on the market, has problems with curves.

You must enter the co-ordinates of your shape as DATA in line 9000. These are relative co-ordinates as used in the DRAW statement and start at the bottom right corner to allow several views to be drawn on screen at once.

A possible shape section is shown in the diagram and the co-ordinates are given. To use this in the program you would produce a DATA line as follows:

9000 DATA −50,0,−10,20, 40,20,20,-40,1000

Note that 1000 is a 'rogue' value to signal the end of the DATA to the computer. Once the required shape DATA has been entered, then the program can be RUN.

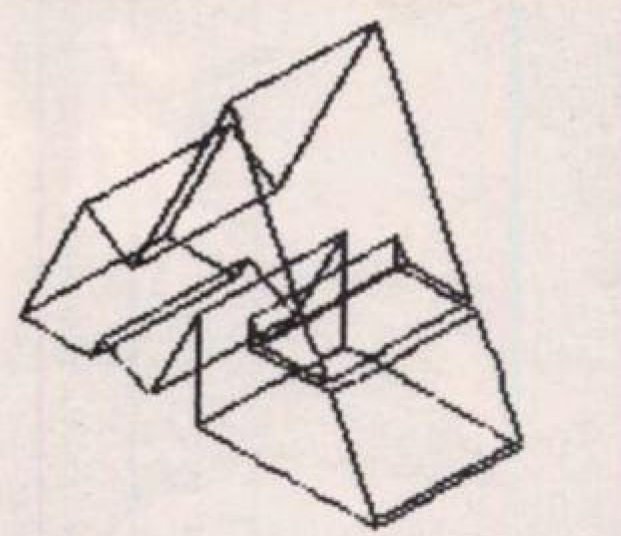
After the title page you are asked to input the depth in pixels of the shape. The program then moves through four separate display modes.

1 ROTATIONAL The shape is moved through 90 degrees in a series of steps specified by yourself.
2 WORKING DRAWING Technically known as 1st Angle Orthographic Projection, the front, side and plan views are drawn in position.
3 OBLIQUE Where all the front facing planes are displayed in full and all the other horizontals are at 45 degrees.
4 ISOMETRIC All sides are displayed equally with all horizontals at 30 degrees.

All the angles in this program are produced by a trigonometrical formula which was carefully and painstakingly worked out by Martin. At each stage of the display, an option to make a copy with the ZX printer is offered.

The DATA included gives the shape that we have used for the screen prints, try running the program with this to get an idea of the way the program works.

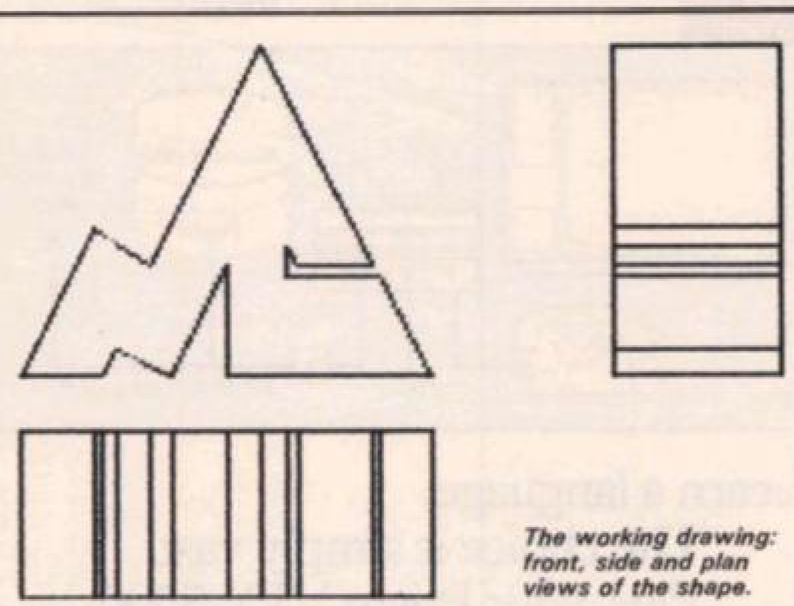
```
   10 BORDER 1: PAPER 1: INK 7: C
LS : RESTORE : GO TO 1000
   40 DIM a(100,2): DIM b(100): L
ET count=0: LET a1=0: LET a2=0:
LET a3=0
   50 READ c: IF c=1000 THEN  GO
TO 61
   51 LET a1=a1+c: IF a1<a2 THEN
 LET a2=a1
   52 IF a1>a3 THEN  LET a3=a1
   60 LET count=count+1: READ d:
LET a(count,1)=c: LET a(count,2)
=d: GO TO 50
   61 LET width=a3-a2
   69 REM * ROTATIONAL *
   70 INPUT "Depth, in pixels (mi
nimum 2,     maximum 50)_ ? ";dept
h: IF depth<2 OR depth>50 THEN
GO TO 70
   72 CLS
   75 PRINT AT 0,0;"Rotational":
INPUT "Number of views (minimum
2) ?"'views: LET views=INT views
: IF views<2 THEN  GO TO 75
   76 LET views=views-1
   77 LET pp=(127+width/2)-a3
   80 FOR n=0 TO PI/2+.001 STEP P
I/(2*views)
   90 FOR m=1 TO count: LET b(m)=
a(m,2)*COS n: NEXT m
   95 LET a=(INT (((PI/2-n)*(90/P
I*2))*100+.5))/100
  100 CLS : PRINT "Angle:";a;" ":
 PLOT pp,70: FOR m=1 TO count: D
RAW a(m,1),b(m): NEXT m: IF n=0
THEN  GO TO 200
  150 LET x=depth*SIN n: PLOT pp,
70: DRAW 0,-x: FOR m=1 TO count:
 DRAW a(m,1),b(m): DRAW 0,x: DRA
W 0,-x: NEXT m
  200 INPUT "Copy ? (y/n) "; LINE
 k$: IF k$="y" THEN  COPY
  220 NEXT n
```



The isometric display of the shape.

Technical

```
 225 REM * WORKING DRAWING *
 230 LET pp=ABS a2+10
 250 CLS : PRINT AT 0,0;"Working drawing"
 260 PLOT pp,70: FOR n=1 TO count: DRAW a(n,1),a(n,2): NEXT n
 270 PLOT 170,70: FOR n=1 TO count: DRAW 0,a(n,2): NEXT n: PLOT 170,70: DRAW depth,0: FOR n=1 TO count: DRAW 0,a(n,2): DRAW -depth,0: DRAW depth,0: NEXT n
 280 PLOT pp,10: FOR n=1 TO count: DRAW a(n,1),0: NEXT n: PLOT pp,10: DRAW 0,depth: FOR n=1 TO count: DRAW a(n,1),0: DRAW 0,-depth: DRAW 0,depth: NEXT n
 290 INPUT "Copy ? (y/n) "; LINE k$: IF k$="y" THEN COPY
 300 REM * OBLIQUE *
 310 LET x=COS (PI/4)*(depth/2): LET pp=120+width/2
 315 CLS : PRINT AT 5,0;"Oblique"
 320 PLOT pp,10: FOR n=1 TO count: DRAW a(n,1),a(n,2): NEXT n: PLOT pp,10: DRAW x,x: FOR n=1 TO count: DRAW a(n,1),a(n,2): DRAW -x,-x: DRAW x,x: NEXT n
 330 INPUT "Copy ? (y/n) "; LINE k$: IF k$="y" THEN COPY
 499 REM * ISOMETRIC *
 500 CLS : RESTORE 9000: PRINT "Isometric"
 520 DIM c(50,2): LET count=0
 525 READ c: IF c=1000 THEN GO TO 600
 530 LET count=count+1: READ d
 540 LET c(count,1)=c*COS (PI/6): LET c(count,2)=d+(-c*SIN (PI/6))
 550 GO TO 525
 610 PLOT 150,0
 620 FOR n=1 TO count: DRAW c(n,1),c(n,2): NEXT n
 630 LET e=depth*COS (PI/6): LET f=depth*SIN (PI/6)
 640 PLOT 150,0: DRAW e,f
 650 FOR n=1 TO count: DRAW c(n,1),c(n,2): DRAW -e,-f: DRAW e,f: NEXT n
 660 INPUT "Copy ? (y/n) "; LINE k$: IF k$="y" THEN COPY
 999 STOP
1000 FOR n=0 TO 7: READ a: POKE USR "a"+n,a: NEXT n: DATA 48,72,72,48,0,0,0,0
1005 PRINT TAB 10; INVERSE 1;" 3-D Plotter "; INVERSE 0
1010 PRINT '"  A program to turn a shape       through 90 degrees, followed by a working drawing, then an         oblique view, and finally an      isometric view."
1020 PRINT '"  Put your shape (straight linesonly) into the DATA line (9000) in the following format: first, all the relative positions as ina DRAW statement; then, the       number 1000. Seperate all the    numbers by commas."
1021 PRINT "  Please enter the first co-ordsrelative to the bottom right      corner. If you want to enter      your own shape, break into the   program now and alter line 9000."
1050 PRINT AT 21,0;"Any key to continue...": PAUSE 0: GO TO 40
9000 DATA -55,0,0,30,-15,-30,-15,7,-4,-7,-22,0,20,40,15,-10,30,60,30,-60,-20,0,-3,5,0,-8,25,0,14,-27,1000
```



The working drawing: front, side and plan views of the shape.

drawing

...and access an
of news, view

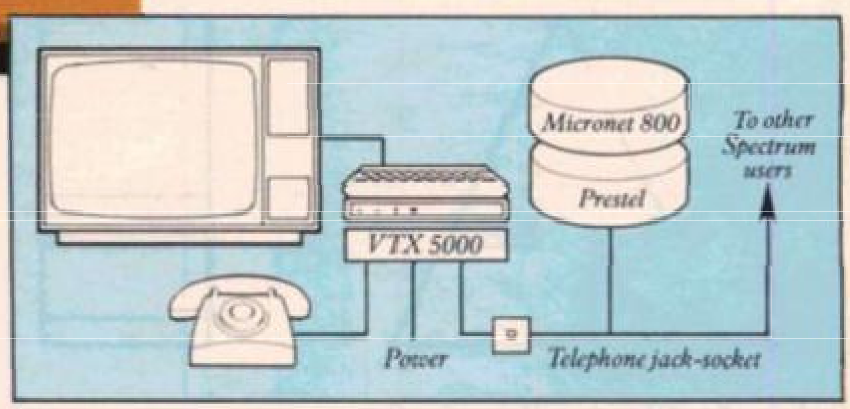




Games



Travel



Simply by connecting your Spectrum to the telephone line, via the award-winning Prism VTX 5000 communications modem, you can get to all the information, games and communication facilities held on viewdata mainframes.

So, you can link into a variety of services, including Prestel and Micronet 800. Access hundreds of software programs – many completely free. Talk to fellow users. Keep up-to-date with all the latest national, international and micro news. Play on-screen games. Check sports results. Book travel and entertainment. Learn a language.

The choice is simply vast.

Plus, your Prism VTX 5000 allows you to communicate directly with other Spectrum owners.

And now, our special offer means that you can do all this for just £11* a month. So complete the coupon and send it to us today, and we will let you have full details of our unbeatable offer.

* All prices quoted refer to domestic use only. For business use the Prestel annual standing charge is £66. Therefore business users will be billed £44.50 per quarter.

exciting world
s, facts & fun!

BRITISH MICROCOMPUTING AWARD
Peripheral
VTX 5000
of the Year.
1984



Swapping



Educational software



Shopping



News



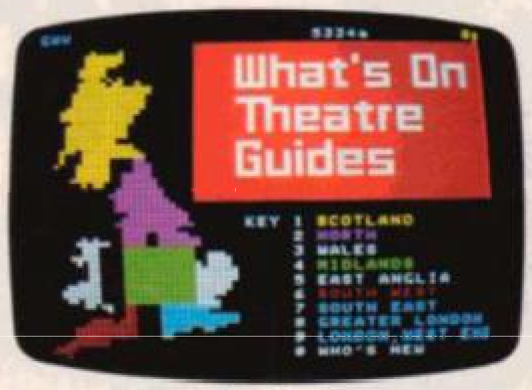
Competitions



Telesoftware



Sport



Booking



Mailbox

FOR JUST £11* A MONTH YOU WILL GET:

- **A Prism VTX 5000 worth £99.95.**
- **One year's subscription to Prestel worth £20.00.**
- **One year's subscription to Micronet 800 worth £32.00.**
- **Free user-to-user software – normally retailing at £19.00**
- **Quarterly hardware and software offers.**
- **Discounted software through Micronet 800 (up to 20% off retail prices).**

Prism Microproducts Ltd.
Prism House, 18-29 Mora Street
City Road, London EC1V 8BT.
Tel: 01-253 2277

PRISM

Prestel and the Prestel symbol are trademarks of British Telecommunications.

Please send me full information on your special offer to Spectrum users.

Name: ______________________

Address: ______________________

____

____

____

ZX6

Spectrum lessons

Mike Edmunds continues his quest against ignorance and assesses some of the latest educational software packages.

Educational software has, to date, been of a very mixed quality, with often only a limited use in the classroom. Now, with the advent of material such as the Dudley Programs, we find software intended for use right across the curriculum.

The programs, all with the theme of travel, have been designed specifically for use in schools by a team of 24 teachers and advisers from Dudley Metropolitan Borough, in conjunction with Five Ways Software.

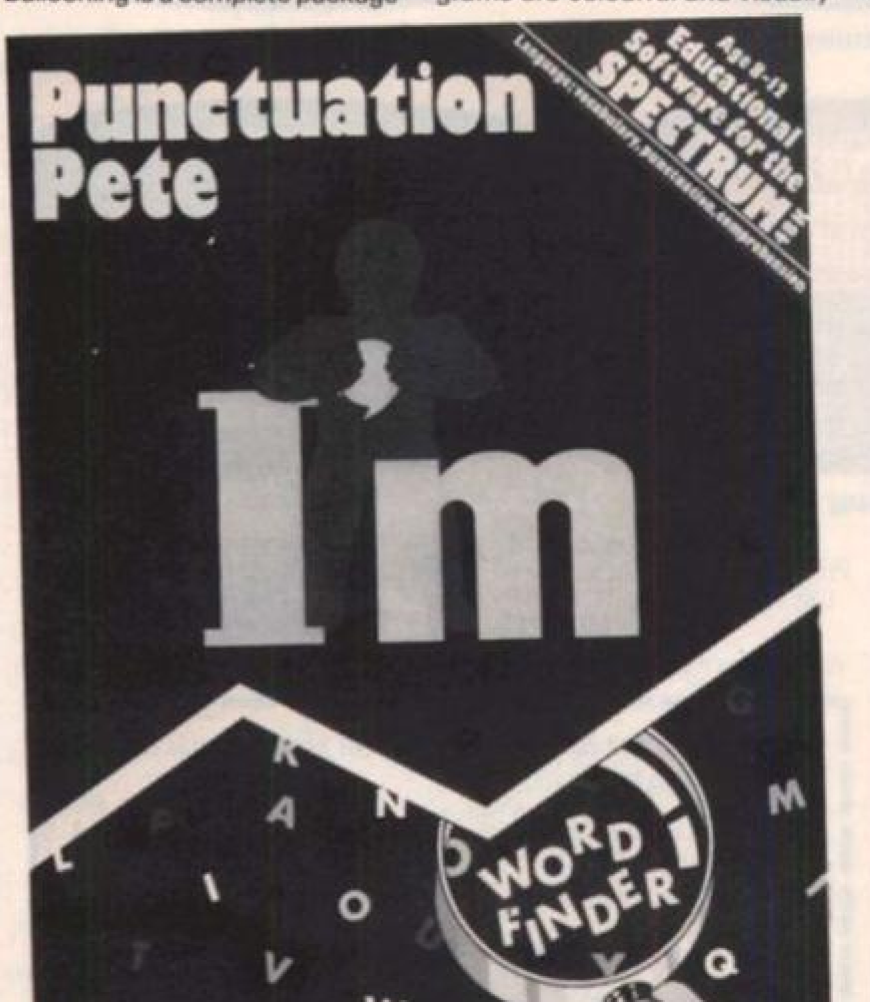
Each of the four titles; Punctuation Pete/Wordfinder, Car Journey, Special Agent and Ballooning is a complete package in itself or can be used in conjunction with the other titles to provide a really comprehensive series of resource materials.

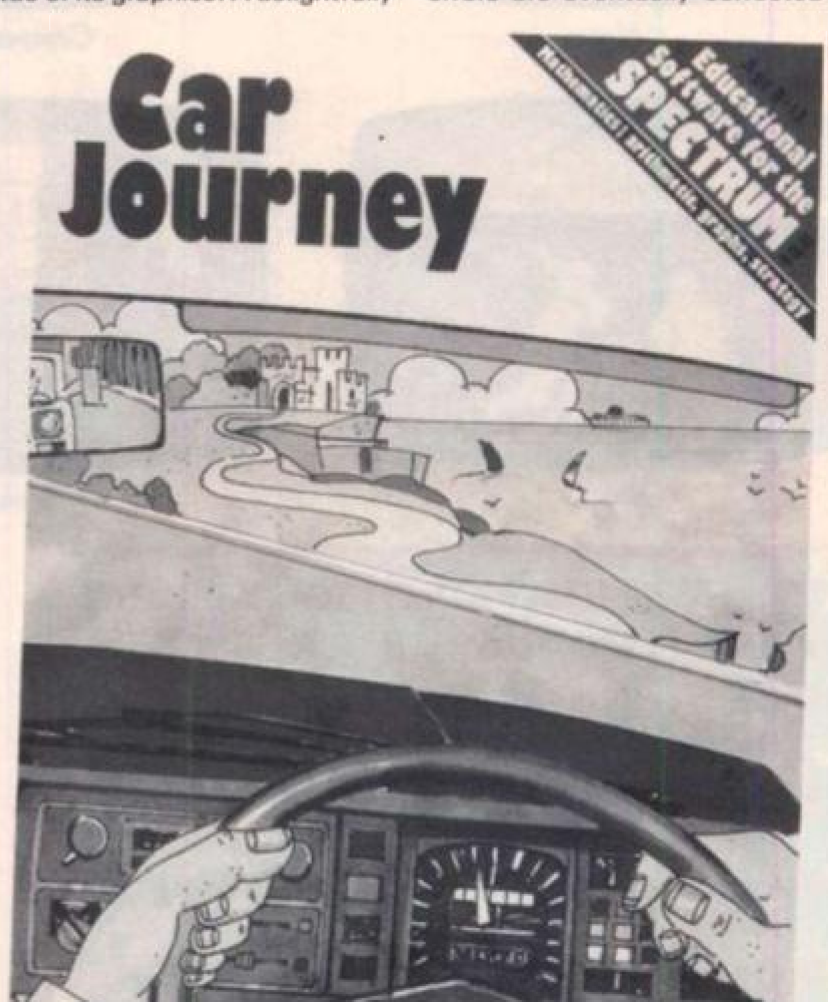
Each package comes in an A4 folder with a detailed guide/workbook which not only outlines the aims of the program but also suggests numerous associated activities to be carried out using either the program or the workbook material. There is also a set of keyboard overlays, a summary of instructions and a 'run card'. Loading is straightforward and the pupil is greeted by an illustrated title screen detailing loading time. As with the workbook, the programs are colourful and visually attractive, good use of graphics being maintained throughout the series.

Punctuation Pete Wordfinder

The first package, Punctuation Pete/Wordfinder, is really two for the price of one. Pete lives inside your computer and it is his job to keep everything tidy. Your job is to help him sort out sentences that have lost their punctuation. Although this type of program has been around for quite a while, this version is a step ahead of the others by virtue of its graphics. A delightfully animated Pete is guided around various texts to correct the mistakes (often physically kicking them into submission it seems!)

This is aimed at reasonably able readers and provides various texts, chosen at random from a large bank, all on the theme of travel. The child can choose his own level, beginning with Junior Scribe (full stops and capital letters). If he can master this level he may wish to aim for higher status, where the exercises are correspondingly more difficult. Mistakes by the child are indicated by Pete and a second try is invited. Continuing errors are eventually corrected





by the computer.

Wordfinder, the second half of the pack, provides an extensive thesaurus on the travel theme, its aim being to develop vocabulary. Words can be selected from a comprehensive list and all linked possibilities then examined. This is a novel idea and it would be nice to think of this program 'sitting' at the back of a classroom available for constant referral as a source of vocabulary, stimulation and ideas, especially for subjects such as creative writing, topic work etc.

Car Journey

The second package, Car Journey, involves travelling around the main roads of England and Scotland and combines geography and mathematics with logical thinking and evaluation. Four levels of skill are provided and the child first has an opportunity to get acquainted with the vehicles and conditions in the Driving School. Here, selection of vehicle, route planning, cost analysis and fuel consumption all prove to be important decisions which the child needs to make. The correct choice makes the journey easy!!! Having passed through the Driving School, the child should be well equipped to tackle the hazards of everyday motoring. As an added bonus here you can also opt for driving with hazards, anything from radar traps to snowstorms!

This program is realistic in approach and the child soon learns to tackle the tasks systematically in order to achieve good results. This is an excellent way of applying decision making and logical thinking to an everyday 'real-life' situation.

Special Agent

In Special Agent you are recruited by M16, your job being to catch an enemy spy. This is a maddeningly frustrating program — sometimes the spy can be tracked down quite quickly, at other times you can spend weeks (or even months) chasing him around the cities of Europe. Continued lack of success might spoil this program for some children but it can also be very addictive! The strategies required are soon discovered and the various countries and cities involved quickly become familiar. (It is a decided advantage to have several atlases on hand!)

The program also gives practice in the workings of train and airline timetables . . . all this whilst trying to understand intelligence reports and decipher codes! As a program for older primary children upwards, this is an ingenious way of encouraging quick, logical, decision making based on a wide variety of information. As an alternative to the normal procedure it is also suggested that the game be played with either limited time or limited money!

Ballooning

Ballooning, the final title in the series, is a simulation program which encourages children to investigate and explore the science of lighter-than-air flight. There are several levels and initial tuition is gained in the Flying School. This is an essential duty before undertaking your Flying Test and eventually your Live Mission. If you pass your test you are assigned a score and your subsequent activities are influenced by the rank you have been given.

You are then ready to embark upon a solo flight with specific tasks to perform. These are not at all straightforward—I'm afraid I left the vet stranded on a hillside miles away from the sick sheep! The program and activities suggested within the package introduce children to various scientific principles and seem an ideal way of combining learning with enjoyment.

To conclude

Taken singly, each of the programs in this series is excellent. Put together, as a complete resource package, they are outstanding — each having the right blend of interest and skill to ensure continued usage. The programs are well written and overall presentation is of a high quality.

As a theme package they test the child's abilities in all areas. Marvellous — fun whilst learning!

Each of the packages reviewed above costs £9.95 and is published by Heinemann Computers in Education Ltd, 22 Bedford Square, London WC1B 3HH.





"THE GAMES ARE GENUINELY EDUCATIONAL IN THE NICEST SENSE OF THE WORD WITHOUT BEING COMPETITIVE OR TRIVIAL."

THE GUARDIAN

"THE RESULT OF A PRODUCTIVE COLLABORATION BETWEEN EXPERIENCED INFANT TEACHERS AND GOOD PROGRAMMERS."

THE TIMES EDUCATIONAL SUPPLEMENT

"...FUN AS CHILDREN LEARN"

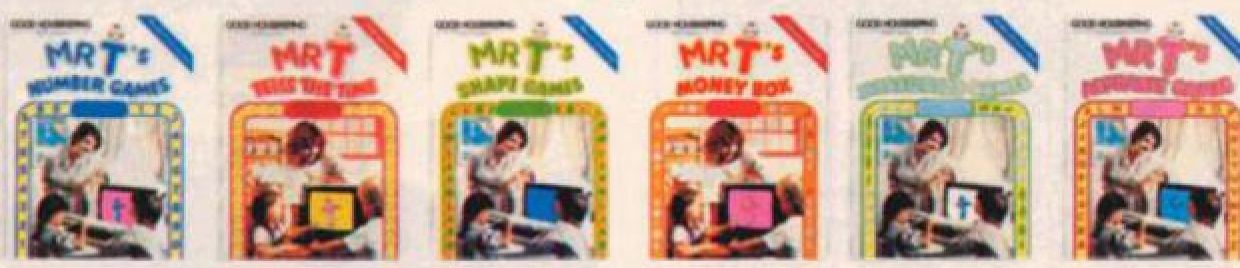
PERSONAL COMPUTER WEEKLY

"THE BEST THING I'VE SEEN IN MY LIFE."

7 YEAR OLD INTERVIEWED BY THE DAILY TELEGRAPH

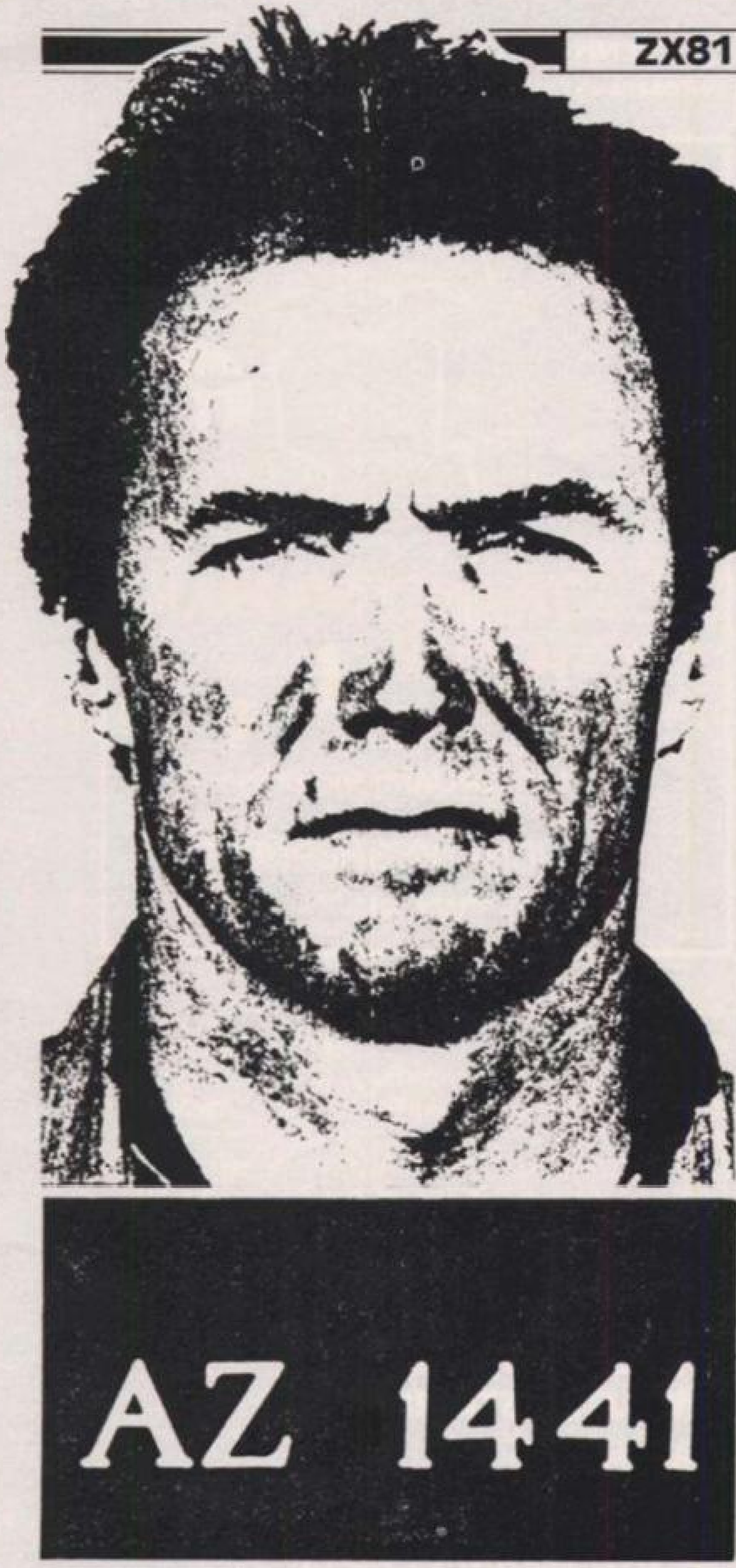
SUCCESS HAS GONE TO OUR PRICE.

~~£12·95~~ £9·95



The highly popular Mr T Educational Software for children 3-7 years.
All now available at £9.95 each and at all good stockists.

GOOD HOUSEKEEPING SOFTWARE



Escape

You've seen the film, now play the game with Lynne Rollinson of Bolton

One of the few graphic adventure games that we have received, Escape provides a real challenge to would be Alcatraz fugitives!

To escape you must first climb the ladder using key '7'. Once at the top you must try and avoid the guard, for, if caught you will be promptly dragged back to the top of the ladder and thrown back to the bottom! This is a very effective routine and I often allowed myself to be caught just so that I could watch it!

Assuming that you eventually scale the ladder and sidestep the guard, you must now make your way along to the top escape hatch and wait until it is in line with the lower hatch. When the two are aligned then jump! If you have judged your leap correctly you are free and your successful escape will be recorded on the display. Oh, one last thing, just to make life a little more difficult you are given a limited amount of time in which to accomplish all this. The amount of time you have left is displayed on a clock on the screen. Note that "time", "escapes" and "0" in line 10 and 20 are INVERSE characters.

This is a very well written program and extra speed was achieved by using POKES to the display file (screen) rather than the command PRINT.

Variables

ES The number of successful escapes the player has made. At the start of a game it is set to 156 (code for an inverse 0) but this is increased for each escape.
P Used as a flag in lines 320 to 400 and is an indicator as to whether the player is on the ladder or not. It is also used to blank out the old asterisk after a move.
A$ Used to erect the ladder.
B Used to alter the extent of the guard's movement, an inverse asterisk.
Z The start of the display file.
X The player's position. X is added to Z to give the current position of the player on the screen.
T The time which remains to the player — set at 165 (code for an inverse 9) at the start of each escape.
F Usually used in FOR/NEXT loops.
G Used either for a flag when the time runs out or if a successful escape has been made. If the time runs out whilst you are on the ladder then you will slide back down; if you have reached the top and both escape hatches are in line then you will have escaped.
L Used in line 815 as a delay.

Along the lines

10-150 Build the scenery and initialise the variables.
160-310 Main program loop. This is left when your time runs out, an escape is made or you are caught.
320-380 If the player is still on the ladder, set P to 22 (loop to show player falling). P=22 if on the ladder, and 0 if jumping down the escape hatch.
420-640 Caught by guard routine, drags player to top of ladder and then jumps to L320.
650-680 Shows when the player has been caught.
710-740 Shows that an escape has been made.
750-770 Increases escape counter, check for end of game.
800-820 Scroll clear screen.
830-980 Introduction.



```
   1 REM ***********************
        *     ESCAPE BY       *
        *   LYNNE ROLLINSON   *
        ***********************
  10 PRINT AT 2,0;"■■■■ ■time■■■
■"
  20 PRINT AT 2, 16;"■■escapes■■C
■■■"
  30 LET ES=156
  40 LET P=0
  50 LET A$=CHR$ 133+"-"+CHR$ 5
  55 LET B=20
  60 FOR F=3 TO 9
  70 PRINT AT F,0;CHR$ 128;AT F,
8;CHR$ 128
  80 NEXT F
  90 LET Z=34+PEEK 16396+256*PEE
K 16397
 100 LET X=675
 110 LET T=165
 120 FOR F=21 TO 2 STEP -1
 130 PRINT AT F,14;A$
 140 NEXT F
 150
 160 FOR F=B TO -B STEP -1
 170 IF PEEK (Z+X)=151 OR PEEK (
Z+X+1)=151 THEN  GO TO 420
 180 IF PEEK (Z+X)=0 OR PEEK (Z+
X)=22 THEN  POKE (Z+X),23
 190 PRINT AT 10,F/4+RND;CHR$ 12
8+" "+CHR$ 128;
 200 IF T<=156 AND X>=15 THEN  G
O TO 320
 210 IF T<156 AND X<15 THEN  GO
TO 330
 220 PRINT AT 1,F;" ";CHR$ 151;"
"
 230 POKE (Z+45),T
 240 IF X>=15 AND X<=675 THEN  P
OKE (Z+X+33),22
 250 IF INKEY$="7" AND X>15 THEN
  LET X=X-33
 260 LET T=T-0.15
 270 IF PEEK (Z+X+33)=0 AND X<>7
08 THEN  GO TO 330
 280 IF X<=15 THEN  LET X=X-(INK
EY$="5")
 290 IF INKEY$="5" AND X<=15 THE
N  POKE (Z+X+1),0
 300 NEXT F
 305 LET B=B-1
 307 IF B=0 THEN  LET B=20
 310 GO TO 160
 320 LET P=22
 330 FOR G=Z+X+33 TO Z+700 STEP
33
 340 POKE (Z+15),0
 350 IF PEEK G=128 THEN  GO TO 6
50
 360 POKE G,23
 370 POKE (G-33),P
 380 NEXT G
 390 POKE (G-33),0
 400 IF P=0 THEN  GO TO 710
 410 LET F=15
 420 FOR D=ABS F TO 14
 430 LET X=15
 440 PRINT AT 1,D;" *";CHR$ 151
 450 NEXT D
 640 IF F<15 THEN  GO TO 320
 650 FOR F=0 TO 10
 660 PRINT AT 0,0;"CAUGHT WHILST
TRYING TO ESCAPE"
 670 PRINT AT 0,0;"
                  "

 680 NEXT F
 690 POKE (Z+708),0
 700 GO TO 800
 710 FOR K=0 TO 10
 720 PRINT AT 0,5;"SUCCESSFUL ES
CAPE"
 730 PRINT AT 0,5;"
  "

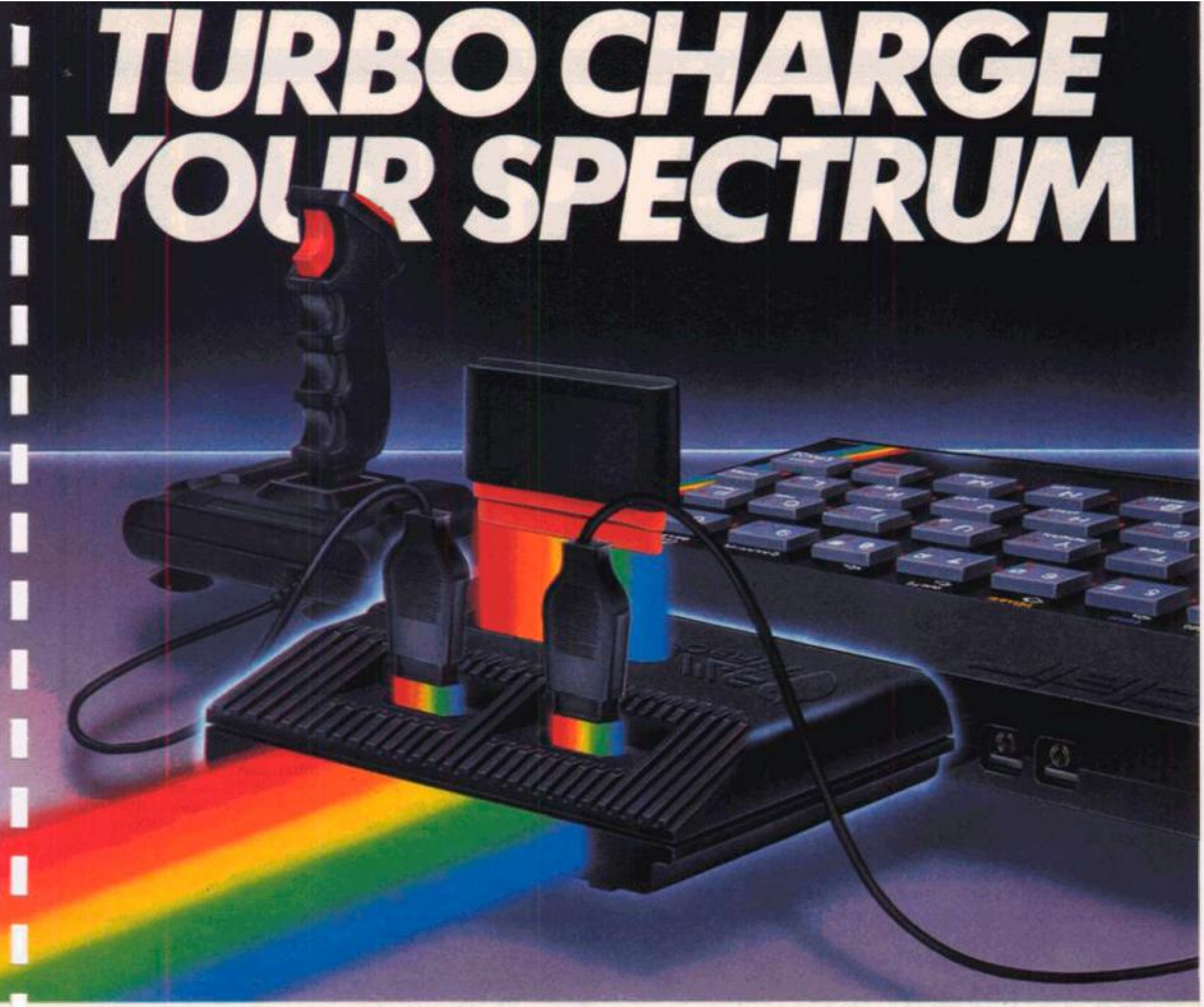
 740 NEXT K
 750 LET ES=ES+1
 760 IF ES=165 THEN  GO TO 920
 770 POKE (Z+60),ES
 780 GO TO 40
 800 FOR F=0 TO 21
 810 REM SCROLL
 815 LET L=RND
 820 NEXT F
 830 PRINT AT 10,6;"PRESS ANY KE
Y TO PLAY"
 840 FOR F=0 TO 15
 850 PRINT AT 10,F;" ";CHR$ 128;
AT 10,F-30;CHR$ 128;" ";
 860 IF INKEY$="" THEN  GO TO 89
0
 870 NEXT F
 880 GO TO 830
 890 PRINT AT 10,0;"
                   "

 900 CLS
 910 GO TO 1
 930 PRINT "YOU HAVE MADE THE MA
XIMUM NUMBER"
 940 PRINT "          OF ESCAPES
"

 950 FOR F=0 TO 100
 960 NEXT F
 970 CLS
 980 GO TO 830
 990 SAVE "GE"
 999 GO TO 1
1000
```


WHEELIE
and
THE
TRAIN
GAME
from
MICROSPHERE
For the Spectrum at £5.95 each.
Available at selected branches of W.H. Smith, Boots, John Menzies and good computer shops everywhere.
(CBM 64 versions due for release soon)

TURBO CHARGE YOUR SPECTRUM



Outperforms any Spectrum interface

The unique Turbo interface from Ram gives you all these features – and more – in one unit:

- A variety of interfaces including: Rom cartridges, two 9-way D plugs for standard joysticks, PLUS full expansion bus at rear.
- Compatible with Kempston and Protek protocols.
- Works with latest Quickshot Mk II auto rapid-fire joysticks!
- Choice of Rom cartridge or tape cassette software.
- Instant program loading with cartridge software.
- Built-in power safety device – unique to Ram Turbo.
- Full one year guarantee.
- Immediate availability – 24 Hr despatch on receipt of P.O./credit card details (cheques – seven days).
- Incredible value – only £22.95.

So don't wait around – simply complete the coupon and send it to us today.

Or call our credit card hot line on 02514 25252. (Access and Visa welcome).

Ram Electronics (Fleet) Ltd, 106 Fleet Road, Fleet, Hampshire GU13 8PA.

24 Hr despatch for credit cards and postal orders.

Please send me:

_____ Spectrum Turbo Interface(s) at £22.95 + £1 p+p (overseas orders £3 p+p)

_____ Quickshot II Joystick(s) at £9.95 (Only when purchased with Turbo – normally £12.95 + £1 p+p)

I enclose cheque/postal order or charge my Access/Visa for: £ _____

Name _____

Address _____

ZX6/84

Tel _____

To: Ram Electronics (Fleet) Ltd, 106 Fleet Road, Fleet, Hampshire GU13 8PA.

RAM TURBO

Trade and export enquiries welcome.

Safe crack

Answer the questions and break into Stephen Parry's Southampton safe.



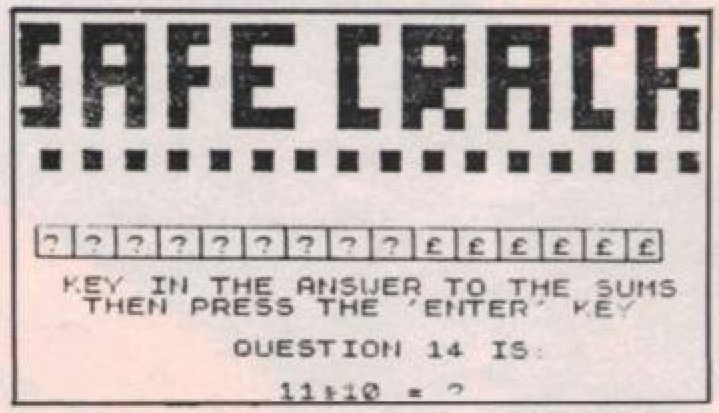
This is not just another code-breaking game but a simple maths program for use in primary schools involving addition, subtraction and multiplication. There are fifteen boxes, each occupied by a red question mark (?) and every correct answer replaces the question mark with a blue pound sign (£). The object is of course to fill all the boxes with pound signs — if anything will attract and occupy the minds of children, it's money! Every wrong answer loses a pound sign but the computer will keep on asking questions until someone wins. Filling all the squares with pound signs will give a large fanfare of sound and a colourful screen display.

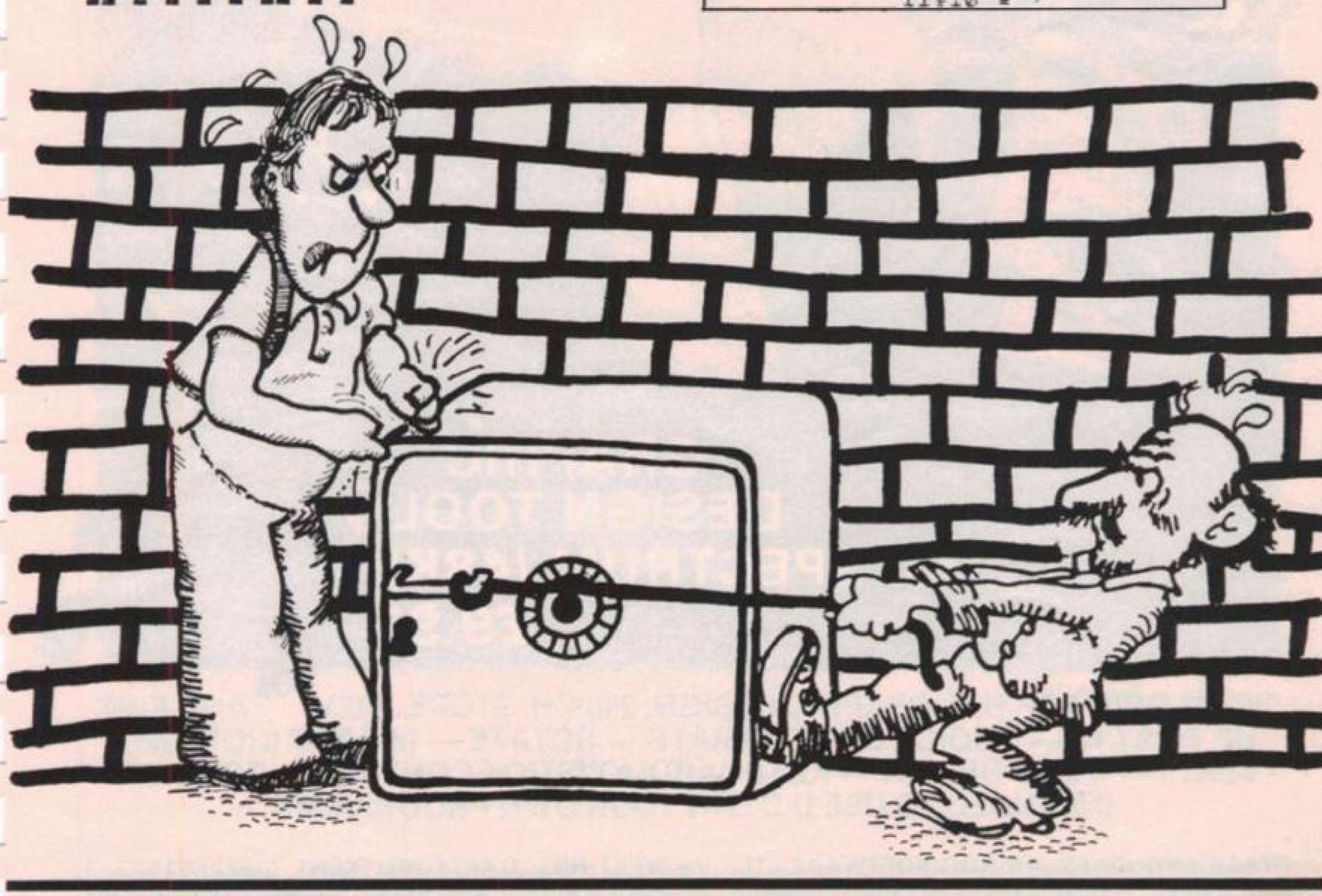
The program runs by first calling the subroutine at 8900. This draws a colourful screen with accompanying sound and a sum is randomly devised in lines 7000 to 7050. To do this, line 7020 chooses whether addition, subtraction or multiplication is to be performed and line 7030 makes sure that if multiplication is chosen then the numbers are kept within the range of 1 to 12. Line 7040 chooses two numbers and the value 100 can easily be changed if smaller numbers are wanted. Line 7050 ensures that no negative numbers result from subtraction and then line 7060 builds the sum itself.

After answering the question from a series of screen prompts, the child is immediately told the answer which is worked out in line 7120. Having compared the child's answer with that of the computer, the program either jumps to a fail message which replaces the pound sign with a question mark or to line 7700, which places a pound sign in the next box. If the variable tot reaches 15, meaning that all the boxes have been filled, the subroutine at line 8000 is called which displays a win message and the usual 'do you want another go' question is asked.

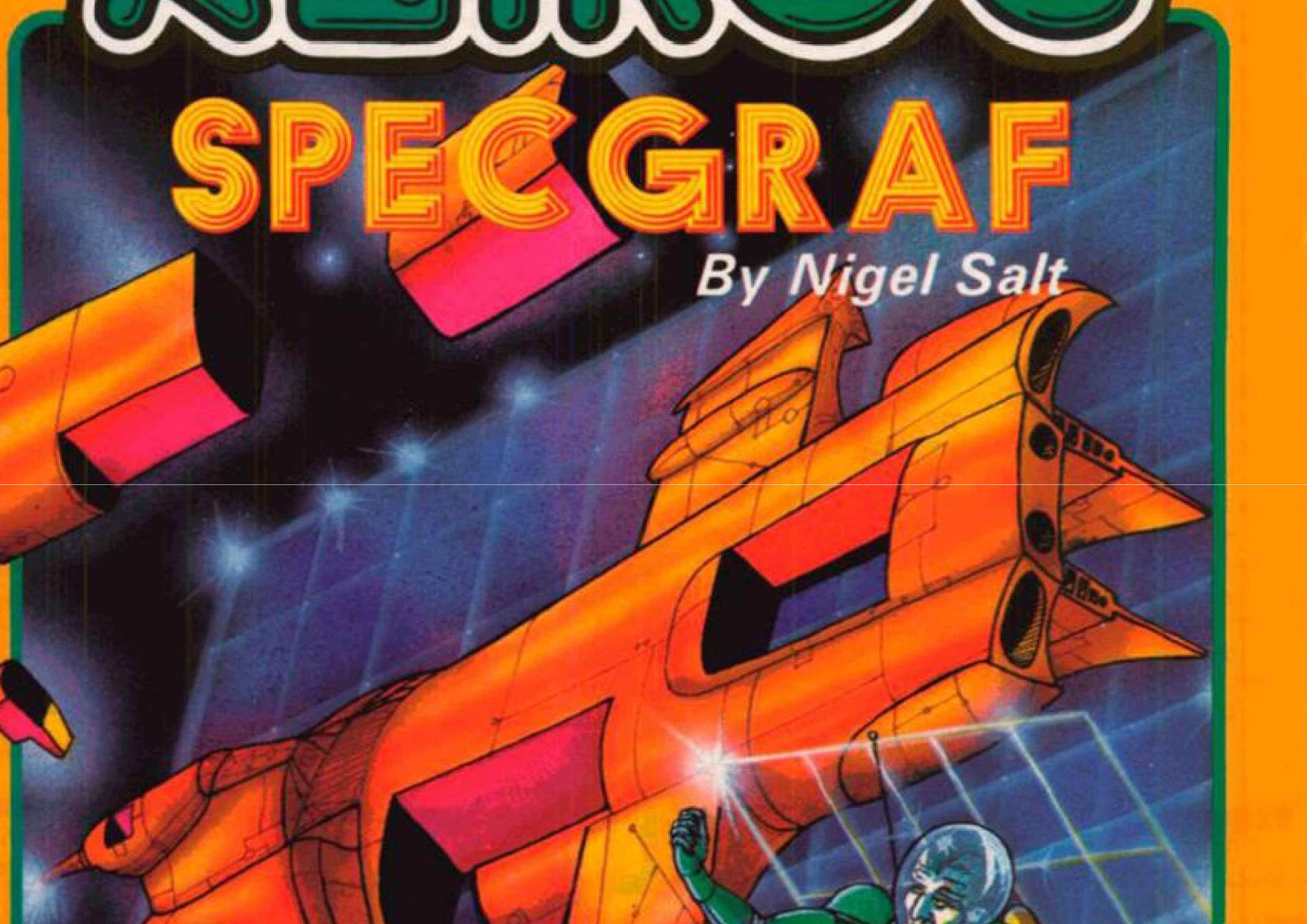
```
  90 RANDOMIZE.
 100 GO SUB 8900
7000 REM CREATE SUM
7005 FOR G=15 TO 21: PRINT  AT G
,0;"
   ": NEXT G
7010 LET F= INT ( RND *3)
7020 LET F$=("*" AND F=0)+("+" A
ND F=1)+("-" AND F=2)
7030 IF F=0 THEN  LET FIRST= INT
 ( RND *12): LET SECOND= INT ( R
ND *12): GO TO 7060
7040 LET FIRST= INT ( RND *100):
 LET SECOND= INT ( RND *100)
7050 IF F=2 AND FIRST<SECOND THE
N  GO TO 7040
7060 LET F$= STR$ FIRST+F$+ STR$
 SECOND
7070 PRINT  AT 15,10;"QUESTION "
;N;" IS:"
7080 PRINT  AT 17,12;F$;" = ?"
7090 INPUT "WHAT IS THE ANSWER ?
";A$
7100 IF  CODE A$<48 OR  CODE A$>
57 THEN  GO TO 7090
7110 PRINT : PRINT  INK 2;"YOUR
ANSWER IS ";A$
7120 PRINT : PRINT "THE RIGHT AN
SWER IS "; VAL F$
7130 IF  VAL A$ <>  VAL F$ THEN
 GO TO 7500
7140 GO TO 7700
7499 REM WRONG ANSWER
7500 PRINT #0; INK 4;"     WRONG
 ANSWER !!"
7510 FOR G=10 TO -10 STEP -1
7520 BEEP .01,G: BEEP .008,G-10
7530 NEXT G
7535 IF TOT<1 THEN  GO TO 7570
7550 PRINT  AT 10,(31-2*TOT); IN
K 2;"?"
7565 LET TOT=TOT-1
7570 LET N=N+1
7590 GO TO 7000
7699 REM CORRECT
7700 LET N=N+1: LET TOT=TOT+1: P
RINT  AT 10,(31-2*TOT); INK 1;"£
": BEEP 1,5
7710 IF TOT=15 THEN  GO TO 8000
7720 GO TO 7000
```

```
8000 REM WIN
8005 FOR W=1 TO 3
8010 FOR G=-5 TO 20: BEEP .01,G:
 BEEP .008,G*2: NEXT G
8020 NEXT W
8030 FOR G=0 TO 21: PRINT  AT G,
0; INK  RND *6;"££££££££££££££££
££££££££££££££££"
8035 NEXT G
8040 INPUT "ANOTHER GO ? (YES OR
 NO)";A$: IF A$="" THEN  GO TO 8
040
8050 IF A$(1)="Y" OR A$(1)="y" T
HEN  RUN
8060 IF A$(1) <> "N" AND A$(1) <
> "n" THEN  GO TO 8040
8800 STOP
8900 BORDER 6: INK 0: PAPER 7: C
LS : PRINT  INK 1;"■■ ■■■ ■■ ■■■
  ■■ ■■■ ■■■ ■■ ■ ■"
8910 PRINT  INK 2;"■  ■ ■ ■  ■
  ■ ■ ■ ■ ■ ■ ■  ■ ■"
8920 PRINT  INK 3;"■■ ■■■ ■■ ■■
  ■  ■■■ ■■■ ■  ■■ "
8930 PRINT  INK 4;" ■ ■ ■ ■  ■
  ■ ■■  ■ ■ ■  ■ ■"
8940 PRINT  INK 5;"■■ ■ ■ ■  ■■■
  ■■ ■ ■ ■ ■ ■■ ■ ■"
8950 PRINT
8960 LET T$="
           "
8970 PRINT  FLASH 1; INK 2;T$
9000 REM SET UP
9010 PLOT 6,98: DRAW 235,0: DRAW
 0,-12: DRAW -235,0: DRAW 0,12
9020 FOR G=20 TO 240 STEP 16
9030 BEEP .1,G/20: PLOT G,98: DR
AW 0,-12
9040 NEXT G
9050 FOR G=1 TO 29 STEP 2
9060 PRINT  AT 10,G;"?"
9070 BEEP .01,G/20: NEXT G
9080 PRINT  AT 12,2;"KEY IN THE
ANSWER TO THE SUMS"'"    THEN PRE
SS THE 'ENTER' KEY"
9090 LET TOT=0: LET N=1: RETURN
```





ANIROG



24 x 24 GRID FOR H.R. GRAPHIC DESIGN. 200 CH. STORE. VIEW AND EDIT UP TO 9 CH. — CHOICE OF 9 FORMATS — ROTATE — MIRROR (HOR. OR VERT.) — MENU DRIVEN — KEYBOARD/JOYSTICK CONTROL — TOOLKIT INCLUDED TO USE U.D.G IN YOUR OWN PROGRAMME

TRADE ENQUIRIES: ANIROG SOFTWARE LTD. 29 WEST HILL DARTFORD KENT (0322) 92513/8
MAIL ORDER: 8 HIGH STREET HORLEY SURREY 24 HOUR CREDIT CARD SALES HORLEY (02934) 6083
PAYMENT BY CHEQUE P.O. ACCESS/VISA

Wall walk



Can you escape from the nasties chasing you? Tat Tang has sent us this superb game.

Based on a well-known arcade game, this program has your little character running along several layers of walls, climbing ladders and generally being chased silly by vicious green meanies. In order to survive you will have to destroy all the aliens by digging pits and luring the unsuspecting nasties into them. Once the alien is in the pit, you must beat him about the head until he dies.

What a gas

Unfortunately, things are not all in your favour — you only have a limited amount of oxygen, so that excessive movement or taking too long will lead to your demise. Needless to say, if you are captured you will lose a life.

Superb graphics, colour and sound all combine to make this an exciting game — it will have you sitting on the edge of your seat, but keep calm and don't p a n i c!

Program details

Type in program 1 to set up the UDG's and RUN it. Now type in the rest of the program and save it by GOTO 9999, the program saves in two sections, one after the other, so don't forget to press a key after the first section has saved. VERIFY both sections by VERIFY """": VERIFY """" CODE and press newline.

An interesting method of keeping the high scores is used, the program POKEs them into the UDG area so that if the program is re-run the high score is kept.

CONTROLS:
- 5 moves left
- 6 moves down
- 7 moves up
- 8 moves right
- 9 Dig left
- 0 Dig right

Make sure that the following lines are entered in graphics mode:
50,60,100,150,160, 170,180,200,213,300,310 380,400,405,410,470,610, 620,640,720,740,780,790, 800,820.

```
10 FOR i= USR "a" TO  USR "a"+167: READ x: POKE i,x: NEXT i
20 DATA 0,24,24,255,189,60,36,102,195,66,126,90,126,126,36,66,255,255,24,24,24,2
4,24,255,195,195,195,255,255,195,195,195,0,28,28,12,60,12,22,50,0,56,56,48,60,48
,104,76,12,16,40,68,66,1,0,0,0,0,1,66,68,40,16,12,96,8,20,34,66,128,0,0,0,0,128,
66,34,20,8,96,0,129,129,129,193,243,235,255,255,129,129,129,129,129,255,255,255,
129,129,129,129,129,129,255,255,24,24,24,24,24,255,255,24,60,126,126,126,126,126
,126,0,124,66,66,124,64,64,0,0,60,66,66,82,74,60,0,0,124,66,66,124,68,66,0,0,60,
```

```
  10 POKE 23693,56: CLS : GO TO 575
  20 CLS : DEF FN R(X)= INT ( RND *X): GO TO 610
  30 LET X1=X: LET Y1=Y
  40 IF  INKEY$ ="H" THEN  GO SUB 750
  50 IF X=3 OR X=8 OR X=13 OR X=18 THEN  LET Y=Y+( IN 61438=251 AND Y<31)-( IN 6
3486=239 AND Y>0): LET A$=("E" AND Y1>Y)+("F" AND Y1<Y): IF Y1 <> Y THEN  LET UN
=UN+.25: GO TO 90
  60 LET UN=UN+.15: LET A$="A"
  70 IF  ATTR (X+1,Y)=57 AND  IN 61438=239 THEN  LET X=X+1: LET F=1
  80 IF  ATTR (X-1,Y)=57 AND  IN 61438=247 THEN  LET X=X-1: LET F=1
  90 PRINT  AT X1,Y1;" "
 100 IF F=1 THEN  PRINT  AT X1,Y1; INK 1;"D"
 110 IF X1 <> X THEN  LET UN=UN+.15
 120 LET F=0: IF  ATTR (X,Y)=57 THEN  LET F=1
 130 IF  IN 61438=255 OR X <> 3 AND X <> 8 AND X <> 13 AND X <> 18 THEN  GO TO 2
10
```

```
 140 LET V= ATTR (X+1,Y+1): LET V1= ATTR (X+1,Y-1): IF V=58 OR V1=58 THEN  LET S
C=SC+5: PRINT  AT 0,6;SC
 150 IF  IN 61438=253 AND  ATTR (X-1,Y-1) <> 57 AND  ATTR (x-1,y-1) <> 60 AND  A
TTR (X,Y-1) <> 57 THEN  LET UN=UN+.15: LET W=1: PRINT  AT X,Y; INK 1;"E": FOR N=
1 TO 5: PRINT  AT X,Y-1;"G": PAUSE 2: PRINT  AT X,Y-1;"H": PAUSE 2: PRINT  AT X,
Y-1;" ": BEEP .01,Y: PRINT  AT X+1,Y-1;"K": NEXT N: PRINT  AT X+1,Y-1;" "
 160 IF  IN 61438=254 AND  ATTR (X,Y+1) <> 57 AND  ATTR (X-1,Y+1) <> 57 AND  ATT
R (X-1,Y+1) <> 60 THEN  LET UN=UN+.15: LET W=2: PRINT  AT X,Y; INK 1;"F": FOR N=
1 TO 5: PRINT  AT X,Y+1;"I": PAUSE 2: PRINT  AT X,Y+1;"J": PAUSE 2: PRINT  AT X,
Y+1;" ": BEEP .01,Y: PRINT  AT X+1,Y+1;"K": NEXT N: PRINT  AT X+1,Y+1;" "
 170 IF V1=58 AND W=1 THEN  PRINT  AT X+1,Y-1; PAPER 6;"C"
 180 IF V=58 AND W=2 THEN  PRINT  AT X+1,Y+1; PAPER 6;"C"
 190 LET W=0
 200 IF  ATTR (X,Y)=60 THEN  PRINT  AT 2,5;"YOU HAVE BEEN MUNCHED!"; AT X,Y; INK
 1; OVER 1; FLASH 1;"A": GO TO 530
 210 IF  ATTR (X,Y)=62 THEN  GO TO 880
 213 PRINT  AT X,Y; INK 1;A$: IF  ATTR (X+1,Y)=56 THEN  PRINT  AT 2,2;"IDIOT YOU
 FELL DOWN A PIT"; AT X,Y;" "; INK 1; AT X+1,Y;"A": GO TO 530
 215 IF XY <= 125 AND F2=0 THEN  GO TO 850
 220 FOR N=1 TO (UN*10)/LV: PLOT  INK 2;XY,8: DRAW  INK 2; OVER 1;0,7: LET XY=XY
-1: LET UN=UN-.1: NEXT N
 230 IF XY <= 72 THEN  PRINT  AT 2,0;"YOU'VE DIED FROM LACK OF OXYGEN": PAUSE 10
0: GO TO 530
 240 LET A1=A(Z): LET B1=B(Z)
 250 IF  ATTR (A1+1,B1)=56 THEN  GO TO 470
 260 IF A1=X THEN  IF A1=3 OR A1=8 OR A1=13 OR A1=18 THEN  GO TO 350
 270 IF C(Z) <> 0 THEN  GO TO 300
 280 IF  ATTR (A1+1,B1)=57 OR  ATTR (A1-1,B1)=57 THEN  GO TO 330
 290 GO TO 350
 300 LET A(Z)=A(Z)+C(Z): IF A(Z)=3 OR A(Z)=8 OR A(Z)=13 OR A(Z)=18 THEN  LET C(Z
)=0: PRINT  AT A1,B1; INK 1;"D": LET A1=A(Z): LET F(Z)=1: GO TO 350
 310 PRINT  AT A1,B1; INK 1;"D": LET A1=A(Z): LET B1=B(Z): GO TO 400
 320 GO TO 400
 330 IF A1<X AND  ATTR (A1+1,B1)=57 THEN  LET C(Z)=1: GO TO 300
 340 IF A1>X AND  ATTR (A1-1,B1)=57 THEN  LET C(Z)=-1: GO TO 300
 350 IF A1=X THEN  LET B1=B1+(B1<Y)-(B1>Y): GO TO 380
 360 LET B1=B1+D(Z)
 370 IF B1 >= 31 OR B1 <= 0 THEN  LET D(Z)=-D(Z)
 380 PRINT  AT A(Z),B(Z);" ": IF F(Z)=1 THEN  PRINT  AT A(Z),B(Z); INK 1;"D"
 390 LET F(Z)=0: IF  ATTR (A1,B1)=57 THEN  LET F(Z)=1
 400 PRINT  AT A1,B1; INK 4;"B"
 405 IF F2=1 THEN  PRINT  AT F5,F4; OVER 1; INK 6;"O"
 410 IF A1=X AND Y=B1 THEN  PRINT  AT 2,5;"YOU HAVE BEEN MUNCHED"; AT X,Y; OVER
1; INK 1; FLASH 1;"A": GO TO 530
 420 IF  ATTR (A1+1,B1)=56 THEN  GO TO 470
 430 LET A(Z)=A1: LET B(Z)=B1
 440 LET Z=Z+1: IF Z >= C1 THEN  LET Z=1
 450 IF  INKEY$ ="H" THEN  GO SUB 750
 460 GO TO 30
 470 PRINT  AT A1,B1;" "; AT A1+1,B1; INK 2;"B": FOR N=1 TO 4: BEEP .1,N: BEEP .
1,5-N: NEXT N
 480 IF C1-1=1 THEN  LET SC=SC+10: GO TO 520
 490 LET A(Z)=A(C1-1): LET B(Z)=B(C1-1): LET F(Z)=F(C1-1)
 500 LET SC=SC+10: LET C1=C1-1
 510 PRINT  AT 0,6;SC: LET Z=1: GO TO 440
 520 PRINT  AT 1,0; FLASH 1; INK 4;"BONUS": FOR N=XY TO 72 STEP -1: BEEP .01,XY/
4: PLOT XY,8: DRAW  INK 2; OVER 1;0,7: LET SC=SC+1: PRINT  AT 0,6;SC: LET XY=XY-
1: NEXT N: FOR N=1 TO 50: BEEP .05,N: BEEP .05,50-N: NEXT N: GO TO 20
 530 IF SC>HI THEN  PRINT  AT 1,0; FLASH 1; INK 2;"CONGRATULATIONS A NEW HI-SCOR
E": FOR N=0 TO 50: BEEP .01,N: BEEP .01,25-N: NEXT N: POKE  USR "U",SC-256* INT
(SC/256): POKE  USR "U"+1, INT (SC/256): GO TO 550
 540 PRINT  AT 0,0;: FOR N=1 TO 11: PRINT  INK 8; PAPER 8; OVER 1; FLASH 1;"
                                                        ": NEXT N: FOR N=1 T
O 2: BEEP .5,-5: BEEP .5,-10: PAUSE 40: NEXT N: FOR N=50 TO 0 STEP -1: BEEP .05,
N: NEXT N
 550 PRINT  AT 11,5;"PRESS ANY KEY TO START"; AT 13,3;"PRESS 'I' FOR INSTRUCTION
```





```
S": IF  IN 254=255 THEN  GO TO 550
 570 GO TO (575 AND  INKEY$  <> "I")+(780 AND  INKEY$ ="I")
 575 INPUT "LEVEL(1-5)"; LINE L$: IF L$>"5" OR L$<"1" THEN  GO TO 575
 576 LET LV= VAL L$*4: GO TO 600
 580 LET N=0: POKE  USR "U",N-256* INT (N/256): POKE  USR "U"+1, INT (N/256)
 590 CLS : PRINT  INK 2;"WRITTEN BY T.TANG"; INK 0; AT 13,3;"PRESS 'I' FOR INSTR
UCTIONS": FOR N=19 TO 21: PRINT  AT N,0; INK 4;"████████████████████████████████
": NEXT N: PRINT  AT 20,3; INK 2;"S P E C T R A L - P A N I C": PRINT  AT 11,5;"
PRESS ANY KEY TO START": PAUSE 0: GO TO 570
 600 CLS : LET LE=0: LET SC=0: LET HI= PEEK  USR "U"+256* PEEK ( USR "U"+1)
 610 CLS : INK 0: PRINT  AT 19,0; PAPER 6;"CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMM";
INK 2; PAPER 7;"OXYGEN>"; INK 0; PAPER 6;"MMMMMMMMMMMMMMMMMMMMMMMMNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNN"
 620 FOR N=4 TO 14 STEP 5: PRINT  AT N,0; PAPER 6;"CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL": BEEP .1,50-N: BEEP .1,N: NEXT N
 630 PAUSE 25
 640 FOR N=3 TO 13 STEP 5: LET Z= FN R(13)+1: FOR F=0 TO 5: PRINT  AT N+F,Z; INK
 1;"D": BEEP .05,N+F: PRINT  AT N+F,31-Z; INK 1;"D": BEEP .05,N+F: NEXT F: BEEP
.1,F: PAUSE 10: NEXT N
 650 PRINT  AT 0,0;"SCORE>";SC; AT 0,16;"HI-SCORE>";HI
 660 LET Y= FN R(31)+1: LET W= FN R(4): IF W=0 THEN  LET X=3
 670 IF W=1 THEN  LET X=8
 680 IF W=2 THEN  LET X=13
 690 IF W=3 THEN  LET X=18
 700 IF  ATTR (X,Y)=57 THEN  GO TO 660
 710 LET F2=0: LET Z=1: DIM D(4): DIM A(4): DIM B(4): DIM C(4)
 720 LET A(1)=3: LET D(1)=1: LET B(1)= FN R(30)+1: PRINT  AT A(1),B(1); INK 4;"B
": BEEP .01,A(1): FOR N=2 TO 4: LET A(N)=A(N-1)+5: LET D(N)=1
 730 LET B(N)= FN R(30)+1: IF  ATTR (A(N),B(N))=57 OR A(N)=X AND B(N)=Y THEN  GO
 TO 730
 740 LET W=0: LET C1=5: PRINT  AT A(N),B(N); INK 4;"B": BEEP .01,A(N): NEXT N: L
ET LE=LE+1: DIM F(4): LET UN=0: LET XY=(30-LE)*8+7: FOR N=9 TO 30-LE: BEEP .01,N
: PRINT  AT 20,N; INK 2;"█": NEXT N: GO TO 30
 750 PRINT  AT 1,12; FLASH 1; INK 2;"GAME"; INK 1;"HELD": IF  IN 254=255 OR  INK
EY$ ="H" THEN  GO TO 750
 760 PAUSE 0: PRINT  AT 1,12;"        ": RETURN
 770 POKE 23658,8: CLS : PRINT  AT 11,9; INK 1; FLASH 1;"LOADING"; INK 2;"LOADIN
G"; AT 12,9;"LOADING"; INK 1;"LOADING": INK 7: LOAD "" CODE  USR "A": INK 0: GO
 TO 580
 780 CLS : PRINT "*INSTRUCTIONS*"'"USE CURSOR KEYS TO MOVE"'"9 AND O TO DIG"'"H
 TO HOLD"'"O EXTRA OXYGEN"; FLASH 1'"DIGGING": FOR N=0 TO 31: PRINT  PAPER 6; AT
 12,N;"C"; AT 13,N;"L": NEXT N
 790 PRINT  AT 11,16; INK 1;"E": FOR N=1 TO 5: PRINT  AT 11,15;"G": PAUSE 2: PRI
NT  AT 11,15;"H": PAUSE 2: PRINT  AT 11,15;" ": BEEP .01,11: PRINT  AT 12,15;"K"
: NEXT N: PRINT  AT 12,15;" "
 800 PAUSE 50: PRINT  AT 5,0; FLASH 1;"ALIEN-10 POINTS": FOR N=1 TO 14: PRINT  A
T 11,N; INK 4;" B": PAUSE 15: NEXT N: PRINT  AT 11,15;" "; AT 12,15; INK 2;"B":
FOR N=1 TO 4: BEEP .1,N: BEEP .1,5-N: NEXT N
 810 PAUSE 50: PRINT  AT 5,0; FLASH 1;"BURY ALIEN-5 POINTS"
 820 PRINT  AT 11,16; INK 1;"E": FOR N=1 TO 5: PRINT  AT 11,15;"G": PAUSE 2: PRI
NT  AT 11,15;"H": PAUSE 2: PRINT  AT 11,15;" ": BEEP .01,11: PRINT  AT 12,15;"K"
: NEXT N: PRINT  AT 12,15; PAPER 6;"C"
 830 PRINT  AT 15,2;"HIT 'I' TO REPEAT INSTRUCTIONS"'' TAB 3;"ANY OTHER KEY TO P
LAY GAME": PAUSE 75
 840 GO TO (780 AND  INKEY$ ="I")+(10 AND  INKEY$  <> "I")+(830 AND  INKEY$ ="")
 850 IF  RND >.7 THEN  LET F2=2: GO TO 220
 860 PRINT  AT 1,10; INK 6; FLASH 1;"EXTRA OXYGEN": LET F4= INT ( RND *32): LET
F6= FN R(4): LET F5=(3 AND F6=0)+(8 AND F6=1)+(13 AND F6=2)+(18 AND F6=3)
 865 IF  ATTR (F5,F4)=57 THEN  GO TO 860
 870 LET F2=1: GO TO 220
 880 PRINT  AT 1,10;"            ": LET F2=2: LET F3= INT ( RND *40)+30: FOR N=1
 TO F3: LET XY=XY+1: PLOT XY,8: DRAW  INK 2;0,7: BEEP .01,XY/5: NEXT N
 890 GO TO 213
9998 STOP
9999 SAVE "PANIC" LINE 770: SAVE "UDG" CODE  USR "A",168
```



Quicksoft

Clive Smith reviews some of the more unusual programs on the market.

Microsoftware Magazine — Infinite Software

You may well ask yourself, why is a magazine reviewing someone else's magazine? No, they haven't got anything to do with Argus Specialist Publications! Microsoftware magazine comes in cassette form and really is not in direct competition with the printed mags so, as we are nice kind people, we decided to review it.

The tape is made for the 16 and 48K Spectrum and there's something in it for everyone. I must admit I do tend to be critical when reviewing but will give credit where credit is due. I found this magazine informative, interesting, fun, educational, graphically exciting and very well formatted. Everything that appears on screen can be run out on your printer and if you are rich and own a Currah speech chip, it will talk to you as well.

The magazine contains such things as games, utility programs, competitions, readers' letters and amazing things that you could do with your Spectrum (in the best possible taste!) In Issue 5 there is the start of a 'basic' training course for the newcomer to computing, and a "Machine Code Corner" for the more experienced amongst us.

It is very well written and an enormous amount of energy is put into each user-friendly magazine. At £2.99 it's reasonably priced.

Infinite Software can be found at 73 Alcester Road, Moseley, Birmingham

Auto Data — ERC Computer Services

Auto Data is produced by E R C Computer Services for those of us who like to tinkle with our cars on a Saturday afternoon. The program helps you diagnose the faults we all seem to have.

On loading you are given a menu which lists 1 Engine; 2 Transmission; 3 Braking system and so on. If you have trouble with your suspension (and who hasn't) you press the corresponding number and it displays a sub-menu. The next menu covers all the faults you can get with suspension. Once you have recognised your fault it then moves to another sub-menu and tells you what to do to rectify it.

What it won't do is tell you how to repair your car. If you're no mechanic then there's no point in buying the tape. If you are a mechanic, you'll probably know all about diagnosis and so it's not for you either! As I said earlier, if you want to know what to do if your car won't start and don't know what bit to look at but would like to try and repair it yourself, then try the tape.

The tape covers all parts of your car but not diesel engines or hydro-pneumatic suspension. It cannot list every fault possible as you would need a mainframe to handle that program, but it covers most of the common faults.

ERC Computer Services are at 53 Thurso Crescent, Menzieshill, Dundee, DD2 4AS.

Cut Your Heating Bills — Brane Software

Ever since the 'Save it' campaign, there have been many ideas on the market to help us reduce our heating bills. Well, it's finally come to your computer via G Timms of Brane Software. This tape is ideal if you are a heating engineer or you are just about to install central heating into your house. Once loaded, the computer will ask you for a lot of details about your house; size of room, thickness of walls, type of floor covering, whether your house is draughty or not and so on. Once you have all these details you then find out the price and type of fuel you intend to use; electricity, coal, gas, wood, etc and feed this into your computer. With all these facts and figures it will then work out your running cost for the winter.

The beauty of this program is, if you were thinking of putting loft insulation into your house, you can work out the advantages in sterling without laying out a lot of money. Nothing in this program has been left to guesswork or chance; as long as you take the time and trouble to feed all the information in correctly (and there's a lot of it) you will find this tape a useful tool. To give you some idea, it has a



list of 38 different building materials your house could be made of! Cut Your Heating Bills comes supplied with a comprehensive guide which tells you exactly what the program is capable of and the benefits you can gain.

Brane Software is at Myrtle Grove, Brane, Sancreed, Penzance TR20 8RE.

Paintbox — Print n Plotter Products

I must admit I've never had so much fun with a utility program as with this one. Paintbox enables you to make up your own UDGs with ease, plot and draw by simple cursor movements, fill in your drawings and save screens for use in your programs. It is written by Joe Gillespie for the 48K Spectrum and is supplied with an easy to follow instruction booklet.

It won't make you an artist overnight but after a couple of hours of loading the tape and reading the instructions I managed to draw this picture. . .



As you can see I'm no Picasso! Paintbox does, however, have more serious applications than drawing funny pictures; as an aid in drawing graphs or in technical drawings and so on.

Once you have loaded the tape, you are presented with a menu. From there you can view the available UDG bank; there are up to 84 user-defined characters in store.

If you don't like the available choice, you can invent your own by calling up the drawing board. This enables you to plot your shape on a 8 × 8 grid by using a cursor. Once made, you then store it in one of the four banks.

Another facility on the menu is "Sketch pad". Here you can practice building up a picture using UDGs before entering them onto your main picture. To build up your picture you first hit the 'Precision Plotter' key. Now the fun bit starts.

You are presented with a blank screen apart from a cursor which can either be moved via the keyboard or a joystick. In the bottom right hand of the screen you are given an X and Y coordinate of the cursor. To draw a line you move the cursor to where you want the line to start and press the PLOT key. Moving the cursor to the place where you want the line to end and pressing the DRAW key will instantly produce your line.

Paintbox has facilities to draw circles and arcs and in radial mode, will return the cursor to the start of your line; ideal if you are drawing a spoked wheel. Once you have outlined your picture, you can shade in any of the parts with different ink colours. Finally, you can put on your UDGs to complete the picture.

If you make a mistake you can edit your last command. As you build up your picture it can be stored in memory and recalled at any time. Your picture can then be stored on tape and reproduced either as a screen$ or as memory bytes. It's possible to store up to five screens and this will leave you with about 6K of memory.

To give you some idea what is possible with this program, have a look at one of the colour adverts in your magazine and take a close look at some of the drawings. With a little practice you will find that it's quite possible to draw pictures of that standard.

The tape costs £7.50 and I think you'll find it's the best £7.50's worth you have ever spent. Print n Plotter products also sell some support hardware to go with it such as key overlays and a jotter pad.

Print n Plotter Products are at Dept ZX, 19 Borough High Street, London SE1 9SE.

1K Corral

This is where we put our brand on the best of the mini masterpieces available!

Although the 1K user is in the minority nowadays, the programs in the 1K Corral should be of interest to all programmers as they demonstrate an economical approach to programming and provide skeleton programs for development. Many of the Spectrum programs we receive (especially 48K) tend to be needlessly lengthy and use a lot of repeated code. In a way it's a pity there is so much memory available but a few lessons in efficiency can be learned here.

All the programs will run as printed but the CHR$ form has been substituted for the ZX graphic characters. If you prefer, substitute the respective graphics for the CHR$, for example: PRINT CHR$ 128
PRINT "inverse space"

Robbery
David Webber

The jewel phantom strikes again! Here's the plan: sneak past the guard and half inch the jewels. Just to check that you've got what it takes, try this simulation first. You are the thief (inverse<) and the guard (inverse O) tries to catch you by patrolling the top and left of the vault (pound sign).

To move, use the cursor keys 5,6,7 and 8 but be warned; if you are caught or stray from the playing area then you will be put back to the beginning and your aim is to get the jewels in the quickest time. David's best is a pretty fast 5; mine is a dismal 9. This is one game where flashing characters are used deliberately to increase the difficulty level!

```
1 LET T=0
2 LET M=10
3 LET N=12
5 LET X=2
6 LET Y=X
7 PRINT AT 10,13;CHR$ 140
8 LET Z=0
9 LET T=T+1
10 PRINT AT X,Y;CHR$ 147;AT X,Y;" "
15 LET Y=Y+(INKEY$="8")-(INKEY$="5")
20 LET X=X+(INKEY$="6")-(INKEY$="7")
35 PRINT AT M,N;CHR$ 180;AT M,N;" "
40 LET Z=Z+1
112 IF X=10 AND Y=13 THEN GOTO 195
113 IF X<2 OR X>21 OR Y<2 OR Y>31 THEN GOTO 2
115 IF M=X AND N=Y THEN GOTO 2
120 IF Z<>4 THEN GOTO 10
125 LET N=25-N
130 LET M=19-M
135 GOTO 8
195 PRINT AT 10,13;CHR$ 147
200 PRINT T;" MINUTES"
```

Chemistry
S O'Duyer

This program calculates the percentage composition by mass for a given formula. For instance, NaCl (common salt) works out to be 39.3% sodium and 60.7% chlorine. The user inputs the symbol of the element or compound and its relative atomic mass (which will have to be looked up in a chemistry text book). Three common compounds are stored for instant use by the computer but if you have 16K and the time, all the other information could be stored.

```
5 PRINT AT 0,0;"ENTER SYMBOL OF 1ST ELEMENT"
10 INPUT A$
15 PRINT AT 0,0;"ENTER RELATIVE ATOMIC MASS (RAM)"
20 INPUT I
25 PRINT AT 0,0;"ENTER SYMBOL OF 2ND ELEMENT OR  COMPOUND"
30 INPUT B$
40 PRINT AT 0,0;"ENTER RELATIVE ATOMIC MASS OF 2ND ELEMENT OR COMPOUND"
45 IF B$="SO4" THEN GOTO 160
50 IF B$="CO2" THEN GOTO 180
60 IF B$="CO3" THEN GOTO 200
70 INPUT K
80 LET P=(I/(I+K))*100
90 PRINT AT 11,2;"PERCENTAGE OF ";A$;" = ";P;
100 LET T=(K/(I+K))*100
120 PRINT AT 13,2;"PERCENTAGE OF ";B$;" =";T;
140 PRINT AT 15,5;"TOTAL PERCENTAGE = ";P+T
150 GOTO 5
160 LET K=96
170 GOTO 80
180 LET K=44
190 GOTO 80
200 LET K=60
210 GOTO 80
```

Catch and Digger
Scott Ford

You know the song, 'Catch a falling ball and put it in your bucket!' That's the general idea although random cross winds do make your life a little more difficult. After each failure press Newline to try again. Warning — once the game has started, the only way of stopping is to catch a ball. Keys 5 and 8 move the bucket.

In Digger, you (inverse X) are trying to escape from the pursuing checked square. If it catches you then you expire. You are given some chance though, you can dig a hole by pressing 0 into which you can try to lure your prosecutor.

You can dig more than one hole but only the most recent is fatal to your enemy. And don't think you can hide in your pit for safety — if you are caught in it then you will have dug your own grave!

```
  5 LET I$=" "+CHR$ 130+CHR$ 13
1+CHR$ 129+" "
  6 CLS
 10 LET B=INT (RND*20)+8
 30 LET X=NOT PI
100 FOR Y=0 TO 11
105 LET B=B+(INKEY$="8")-(INKEY
$="5")
110 LET X=X+INT (RND*5)
115 PRINT AT 21,B;I$
130 PLOT X,39-32*(Y/10)**2
140 NEXT Y
150 IF ABS (B*2)-X<4 THEN PRINT
"YOU WIN";Z
160 PRINT "YOU LOST";AT 5,5;"TR
Y AGAIN"
170 INPUT I$
180 RUN
```

```
  10 LET X=NOT PI
  20 LET Y=X
  30 LET A=10
  40 LET B=15
  42 LET C=NOT PI
  45 LET D=C
  50 LET X=X+(INKEY$="6")-(INKEY
$="7")
  60 LET Y=Y+(INKEY$="8")-(INKEY
$="5")
  65 PRINT AT X,Y;CHR$ 189;AT C,
D;CHR$ 52;AT A,B;CHR$ 136
  70 LET A=A+(X>A AND RND<.3)-(X
<A AND RND<.3)
  80 LET B=B+(Y>B AND RND<.3)-(Y
<B AND RND<.3)
  90 IF INKEY$="0" THEN GOSUB 10
00
 100 PRINT AT X,Y;" ";AT C,D;" "
;AT A,B;" "
 110 IF X=A AND Y=B THEN PRINT "
YOU LOSE";Z
 120 IF A=C AND B=D THEN PRINT "
YOU WIN";Z
 200 GOTO 50
1000 LET C=X
1010 LET D=Y
1020 RETURN
```

Back to base
David Webber

This program will convert whole numbers to the corresponding value in a number that has been selected. If the base is less than 10, the numerals are printed consecutively; if greater than 10 then line 50 provides suitable spacing.

Obviously very high and silly inputs will cause much confusion — what do you expect from 1K?

```
  1 PRINT "X=NUMBER TO CONVERT,
Z=BASE."
  2 LET N=1
  3 PRINT AT N,0;"NUMBER?"
  5 INPUT X
  6 PRINT AT N,15;X
  7 PRINT "BASE?"
  8 INPUT Z
  9 LET Y=29
 10 LET T=X/Z
 20 LET S=X-(INT T)*Z
 30 PRINT AT N+2,Y;S
 35 PRINT AT N+1,18;"TO BASE ";
Z;"="
 40 LET X=INT T
 45 IF X=0 THEN GOTO 480
 50 LET Y=Y-1-(2 AND Z>10)-(Z>1
00)
 60 GOTO 10
480 LET N=N+4
500 GOTO 3
```

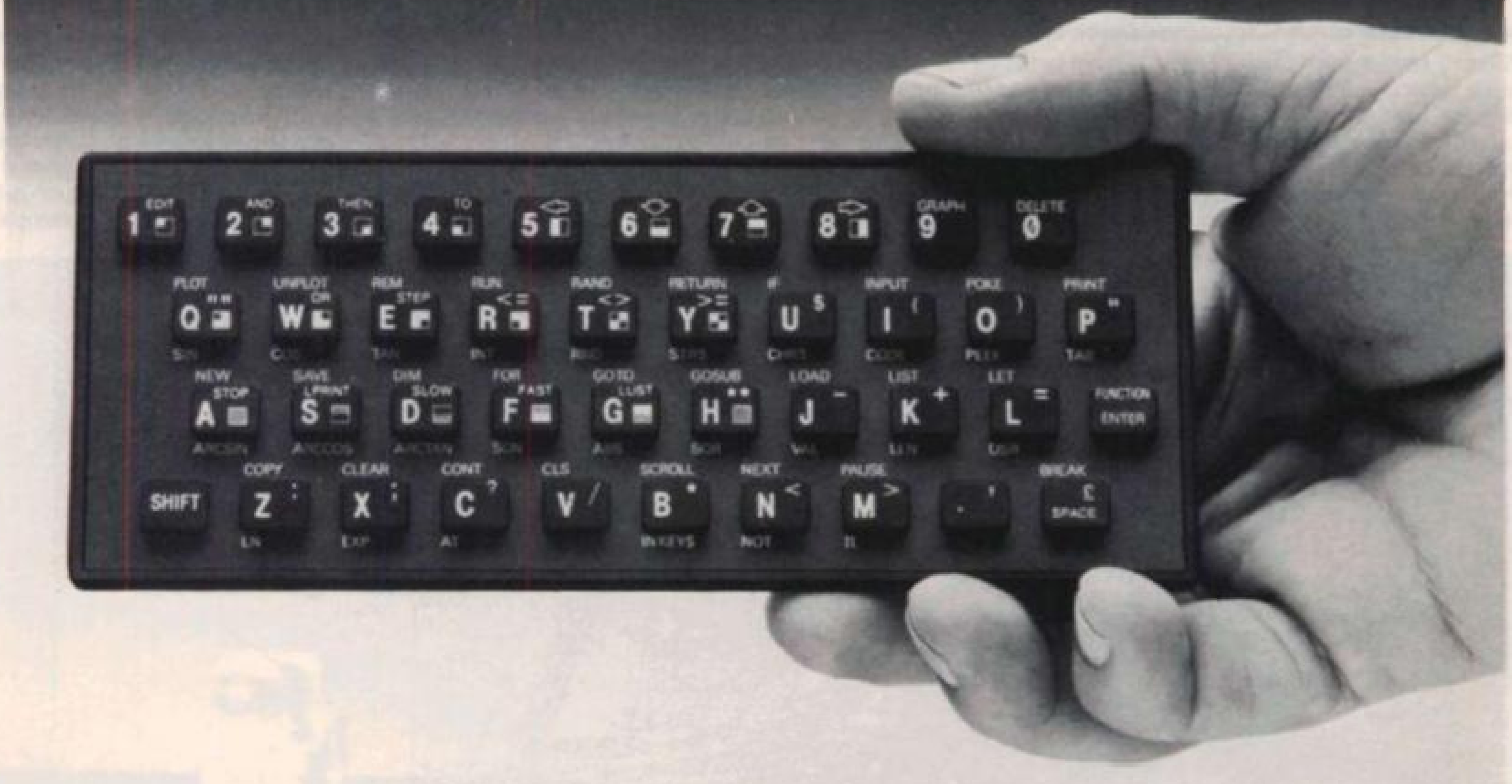
Outer Space
Ian Rogers

Deep in the far reaches of space, a solitary spaceship is in peril. Yes, another dodge-the-scrolling-blobs-type game, but there are some first rate techniques used in this version; going into fast before clearing the screen for instance!

I also liked Ian's concept of "sticky stars" which you have to dodge by moving left and right with keys 5 and 8 and there is even a score and high score.

```
  5 LET S=0
 10 LET A=-9
 20 LET B=VAL "5"
 30 PRINT AT 17,RND*11;"*";AT 1
7,RND*11;"*";AT 6,B;" "
 40 SCROLL
 50 LET B=B+(INKEY$="8" AND B<1
1)-(INKEY$="5" AND B>0)
 60 PRINT AT 6,B;"V";AT 7,B;
 70 IF PEEK (PEEK 16398+256*PEE
K 16399)=23 THEN GOTO 100
 80 LET A=A+1
 90 GOTO 30
100 IF A>S THEN LET S=A
105 PRINT AT 3,16;"SCORE";AT 4,
17;A,,,,TAB 16;"HIGH";TAB 16;"SC
ORE";TAB 17;S,,,,"PRESS ANY KEY"
,,"FOR A NEW GAME"
110 PAUSE 4E4
115 FAST
120 CLS
125 SLOW
130 GOTO 10
```

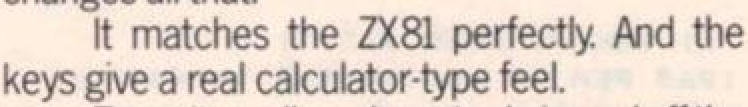
A NEW, IMPROVED ZX81 KEYBOARD AT THE SAME OLD PRICE. £9.95.



There's only one thing wrong with the ZX81. Its keyboard.

Or rather its lack of one.

Since it's flat your fingers don't feel as if there's any response to the pressure put on the keys.



In other words, you're not quite sure which keys you've pressed until the screen actually tells you.

Our new, improved push button keyboard changes all that.

It matches the ZX81 perfectly. And the keys give a real calculator-type feel.

To set it up all you have to do is peel off the adhesive backing and stick it on top of the ZX81 touchpad.

Because no tampering or soldering is involved the guarantee is not affected. And it will last for up to 3½ million operations.

But our keyboard doesn't just come loaded with features. With it comes a separate overlay and a set of coloured stick-on labels to make game playing easier.

It's yours for the original price of £9.95.

Whichever way you look at it, we think you'll agree that it's a keyboard that's quite outstanding.

Orders to Filesixty Ltd., FREEPOST, London W9 2BR.
Cheques/PO made payable to Filesixty Ltd.
Please send me____(qty.) Keyboards at £9.95 each (including VAT and P&P).

Total £____

Name____ BLOCK CAPITALS

Address____

C2

FILESIXTY

Filesixty Ltd., 25 Chippenham Mews, London W9 2AN, England. Tel: 01-289 3059. Telex: 268 048 EXTLDN G 4087.

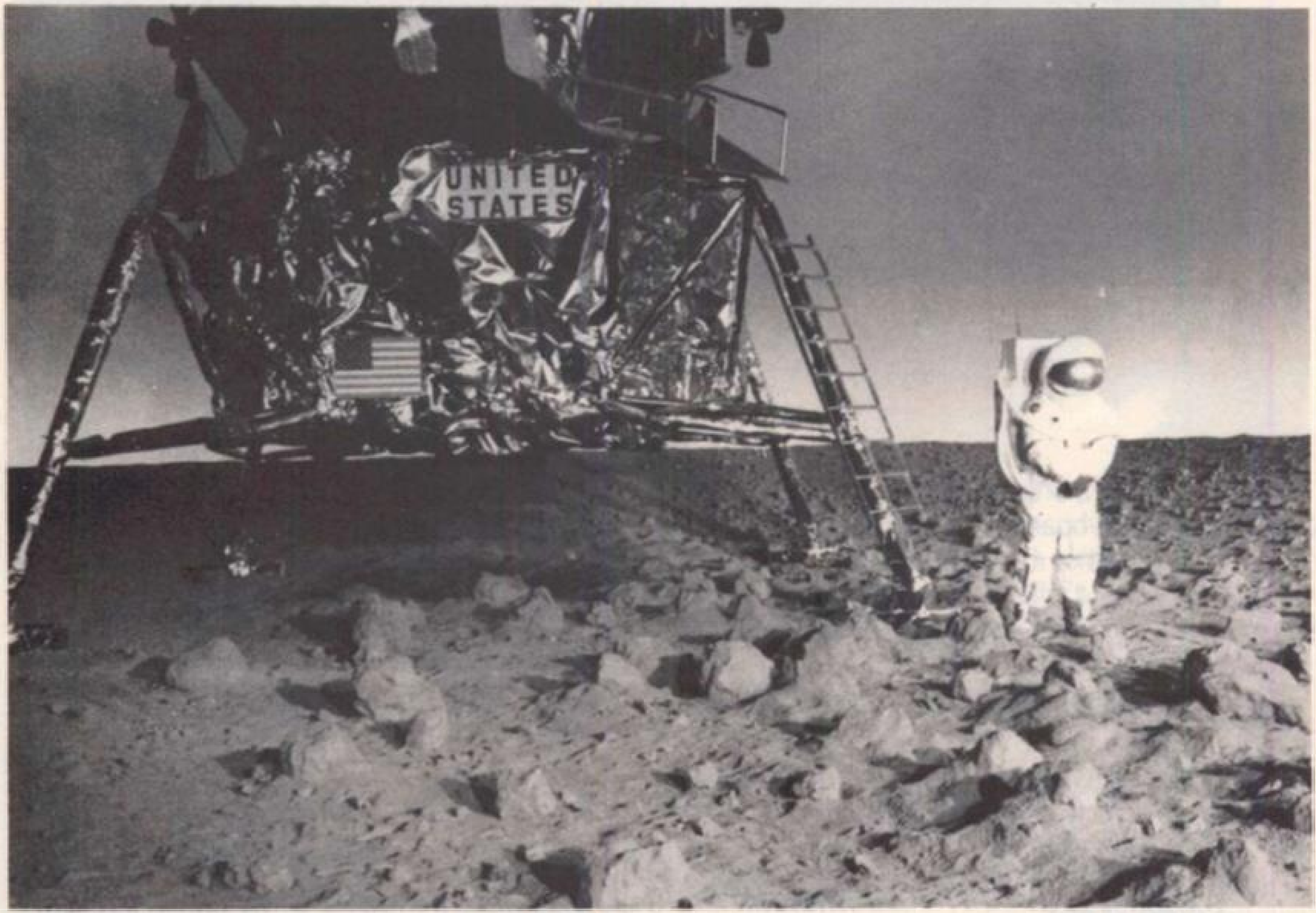
Touchdown

A lander game that will improve your programming from Mike Richardson

When I talked with Durell Software recently, Robert White explained that their first release, Lunar Lander for the Oric, was written in BASIC with the intention that purchasers *should* break into it. It was also REMmed to the hilt to enable the user to understand how the game was constructed.

It didn't take a lot of persuasion to get the other leading light and main Spectrum programmer of Durell, Mike Richardson, to provide us with a Spectrum version. Mike wrote the Spectrum version of Durell's best seller, Harrier Attack, created Jungle Trouble and produced their new winner, Scuba Dive.

So here we are privileged to present a program which demonstrates how the professionals would go about writing a BASIC program. Of course, the software Durell market is all in machine code, but Mike says that he enjoyed the exercise of writing in BASIC again. About half the program is REMs, you can leave these out if you wish and still have a great game to play, but the main intention is for beginners to learn by example.



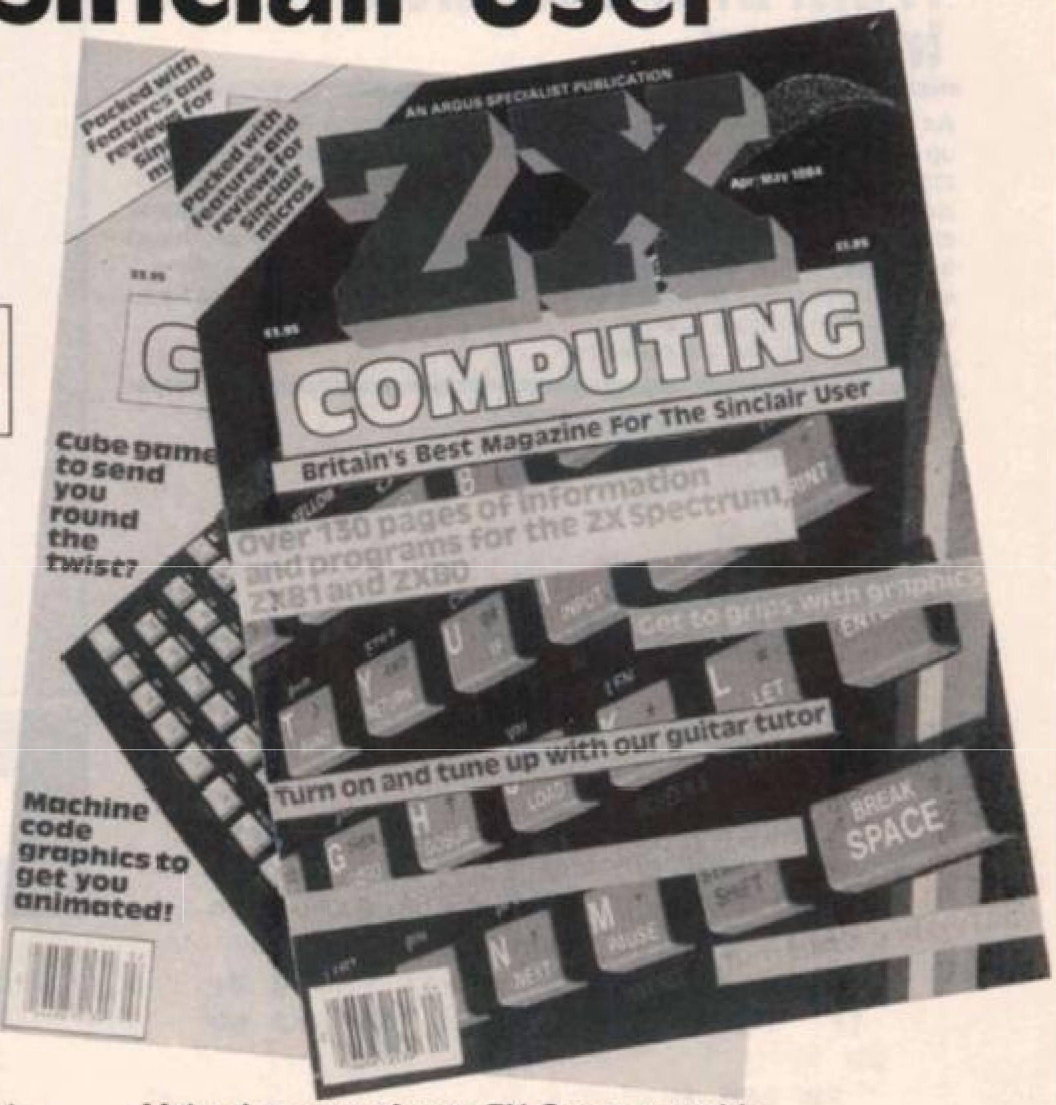

```
  90 REM initial entry point
 100 GO TO 3000
1000 REM if LM is on screen then
 delete it
1010 IF NOT s THEN  GO TO 1080
1020 REM delete LM
1030 PRINT AT y,INT x; INK 7; OV
ER 1;" ";
1040 REM if power was on then de
lete flame
1050 IF t THEN  PRINT AT y+1,INT
 x; INK 7; OVER 1;" ";
1060 REM if horizontal thrust on
 then delete flame
1070 IF e THEN  PRINT AT y,INT x
+e; OVER 1; INK 7; PAPER 8;CHR$
(148-(e+1)/2)
1080 REM is LM now on screen
1090 IF z<0 THEN  LET s=0: GO TO
 1200
1100 REM has LM crashed
```

```
1110 IF z>20 OR ATTR (z,INT w)<>
79 THEN  GO TO 4200
1120 REM has LM landed
1130 IF z=m AND INT w=l THEN  GO
 TO 4000
1140 REM print LM
1150 LET s=1: PRINT AT z,INT w;
OVER 1; INK 4;" ";
1160 REM if thrust is on print f
lame
1170 LET t=0: IF p AND ATTR (z+1
,INT w)=79 THEN  PRINT AT z+1,IN
T w; OVER 1; INK 2;" ";: LET t=1
1200 REM record new LM position
get key press
1210 LET x=w: LET y=z: LET b=COD
E INKEY$
1220 REM test for new thrust
value if fuel still available.
Print the new thrust value.
1230 IF f AND b>47 AND b<59 THEN
  LET p=b-48: PRINT AT 16,27; PA
PER 0; INK 7;CHR$ b;
1240 REM reset horizontal fuel
usage and flame flag, test for
left thrust and record fuel use,
set flame flag and print flame.
1250 LET g=0: LET e=0: IF b=113
AND f THEN  LET d=d+.125: LET g=
2: IF s THEN  LET e=-1: PRINT AT
 y,INT x-1; OVER 1; INK 2; PAPER
 8;" ";
1260 REM test for right thrust
record fuel use, set flame  flag
 and print flame.
1270 IF b=112 AND f THEN  LET d=
d-.125: LET g=2: IF s THEN  LET
e=1: PRINT AT y,INT x+1; OVER 1;
 INK 2; PAPER 8;" ";
1280 REM calculate new x coordin
ate, reduce effect of horizontal
thrust and test that LM is still
within border.
1290 LET w=w+c+d: LET d=d-d/4: I
F w<1 OR w>24 THEN  LET w=x
1300 REM take account of fuel in
 thrust value, calculate new
height and velocity; h is the
printed height,z is the screen
position.
1310 LET p=p AND f: LET v=v+1.5*
(p-5): LET h=h+v/10: LET z=21-IN
T (h/10)
1320 REM calculate fuel usage
1330 LET f=f-p-g: IF f<=0 THEN
LET f=0: PRINT AT 16,27; PAPER 0
; INK 7;"0";AT 12,27;"0 ";
1340 REM display new values
1350 PRINT AT 4,27; PAPER 0; INK
 7;INT h;" ";AT 8,27;INT v;" ";A
T 12,27;f;" ";
1360 REM next move
1370 GO TO 1000
3000 REM print instructions, set
 up graphics and initialise rand
om number generator
3010 GO SUB 9000: GO SUB 9500: R
ANDOMIZE : BRIGHT 1
3100 REM choose landing pad posi
tion
3110 LET l=INT (RND*21)+2
3120 REM set up screen
3130 BORDER 0: PAPER 1: INK 7: C
LS
3140 REM make a black border
3150 FOR a=0 TO 21: PRINT  PAPER
 0;" ";: PRINT TAB 25; PAPER 0;"
       ";: NEXT a
3160 REM plot random stars
3170 FOR a=1 TO 30: PLOT RND*192
+8,RND*80+95: NEXT a
3180 REM draw a random moonscape
3190 INK 0: LET b=5
3200 FOR a=8 TO 200: PLOT a,0: D
RAW 0,b
3210 REM keep landing site flat
and record height of landing pla
tform
3220 IF INT (a/8)=l THEN  LET m=
b: GO TO 3250
3230 REM randomly shape the moon
scape
3240 LET b=b-1+INT (RND*3): IF b
<1 THEN  LET b=1
3250 NEXT a
3260 REM calculate landing posit
ion and print landing platform
3270 LET m=20-INT (m/8): PRINT A
T m+1,l; INK 6;" ";
3280 REM print instruments
3290 PRINT AT 2,26; PAPER 0; INK
 6;"HEIGHT";AT 6,26;"SPEED";AT 1
0,26;"FUEL";AT 14,26;"POWER"; IN
K 7;AT 16,27;"0";
3300 REM initialise game variabl
es
3310 REM variable use is as foll
ows
3320 REM a....FOR/NEXT counter
3330 REM b....Miscellaneous
3340 REM c....Strength of lunar
              wind
3350 REM d....Horizontal thrust
3360 REM e....Horiz thrust flag
3370 REM f....Fuel
3380 REM g....Fuel usage of hor-
              izontal thrust
3390 REM h....LM height
```

```
34ØØ REM l....Landing position
          x coordinate
341Ø REM m....Landing position
          y coordinate
342Ø REM p....Vertical thrust
          power
343Ø REM s....LM on screen flag
3435 REM t....Thrust flame flag
344Ø REM v....LM vertical
          velocity
345Ø REM w....New LM x coordinat
e
346Ø REM x....Last LM x coordina
te
347Ø REM y....Last LM y      "
348Ø REM z....New LM y       "
349Ø LET d=Ø: LET e=Ø: LET s=Ø:
LET t=Ø: LET h=21Ø: LET z=Ø: LET
 v=-1ØØ: LET f=1ØØØ: LET p=Ø: LE
T w=RND*24+1: LET c=(RND-.5)/3
35ØØ REM sound start of game
351Ø FOR a=Ø TO 4: BEEP .Ø2,-1Ø:
 BEEP .Ø1,Ø: BEEP .Ø2,1Ø: BEEP .
Ø3,2Ø: NEXT a
352Ø REM begin play
353Ø GO TO 1ØØØ
4ØØØ REM safe landing ?
     i.e. is landing slow
4Ø1Ø IF v<=-5 THEN  GO TO 42ØØ
4Ø2Ø REM print the landed LM and
play the victory tune
4Ø3Ø PRINT AT z,INT w; INK 6;" "
;: RESTORE 8Ø5Ø: GO SUB 8ØØØ: GO
 TO 43ØØ
42ØØ REM LM crashed
42Ø5 REM ensure crash is on
         screen
421Ø IF z>2Ø THEN  LET z=21
422Ø REM print crashed LM
423Ø PRINT AT z,INT x; FLASH 1;
INK 2;" ";
424Ø REM play something morbid
425Ø RESTORE 81ØØ: GO SUB 8ØØØ
43ØØ REM print instructions to
bottom of screen
431Ø PRINT #1;"Press any key to
play again.";
432Ø REM wait for key press then
 start new game
433Ø PAUSE Ø: GO TO 31ØØ
8ØØØ REM *********************
          Subroutines
         *********************
8ØØ5 REM play a tune
8Ø1Ø READ a: IF a=999 THEN  RETU
RN
8Ø2Ø READ b: BEEP a/1Ø,b: GO TO
8Ø1Ø
8Ø5Ø REM you win tune data
8Ø6Ø DATA 2,7,2,7,1,9,1,12,1,11,
1,9,2,14,2,14
8Ø7Ø DATA 1,14,1,16,1,11,1,12,2,
9,2,9,1,9,1,12,1,11,1,9,2,7
8Ø8Ø REM end of tune
8Ø9Ø DATA 999
81ØØ REM moon wins tune data
811Ø DATA 6,-24,4,-24,2,-24,6,-2
4,4,-21,2,-22,4,-22,2,-24,4,-24,
2,-25,6,-24
812Ø REM end of tune
813Ø DATA 999
9ØØØ REM *********************
9ØØ5 REM user defined graphics
          6 characters are used
9Ø1Ø REM A....Landing platform
9Ø2Ø REM B....LM
9Ø3Ø REM C....Rocket flame
9Ø4Ø REM D....Right thrust flame
9Ø5Ø REM E....Left thrust flame
9Ø6Ø REM F....Crashed LM
9Ø65 REM *********************
9Ø7Ø RESTORE 9Ø8Ø: FOR a=USR "a"
 TO USR "a"+6*8-1: READ b: POKE
a,b: NEXT a: RETURN
9Ø75 REM graphics data
9Ø8Ø DATA 255,24,24,24,24,36,66,
129
9Ø9Ø DATA 24,126,219,255,126,24,
36,66
91ØØ DATA 24,16,24,4Ø,28,28,24,8
911Ø DATA Ø,12,183,238,24,Ø,Ø,Ø
912Ø DATA Ø,24,119,237,48,Ø,Ø,Ø
913Ø DATA Ø,Ø,16,24,247,236,88,1
6Ø
95ØØ REM *********************
95Ø5 REM playing instructions
95Ø7 REM *********************
951Ø BORDER 1: PAPER 1: INK 5: C
LS
952Ø PRINT AT 1,1Ø;"LUNAR LANDIN
G";TAB 1Ø;"------------"''
953Ø PRINT "    The object of th
e game is to"'"safely land your
lunar module on"'"the landing pl
atform. Control"'"your descent b
y pressing keys Ø"'"to 9 to adju
st  your  rocket"'"thrust."''"
  Beware of the lunar wind"'"whi
ch causes lateral movement of"'"
the lunar module";
954Ø PRINT ". Control left"'"and
 right  thrust  with  the 'q'"'"
and 'p' keys respectively."''''
"  Press any key to begin play."
;
9545 REM wait for keypress and
return
955Ø PAUSE Ø: RETURN
```

Britain's Best Magazine For The Sinclair User

ZX COMPUTING



Don't be disappointed — make sure you get every issue of *ZX Computing!*

Just £12.70 will ensure that the next six issues will be lovingly wrapped and posted to you. Just fill in the form below, cut it out and send it with your cheque or postal order (made payable to ASP Ltd) to:

ZX Computing Subscriptions,
Infonet Ltd,
Times House,
179 The Marlowes,
Hemel Hempstead,
Herts HP1 1BB,

Alternatively you can pay with your Access or Barclaycard: simply fill in your card number, sign the form and send it off. Do NOT send your card!

Make the most of your ZX Computer with *ZX Computing* — bi-monthly!

Cut out and SEND TO:

ZX COMPUTING

ZX Computing Subscriptions,
Infonet Ltd,
Times House,
179 The Marlowes,
Hemel Hempstead,
Herts HP1 1BB,

SUBSCRIPTION RATES

(tick ☐ as appropriate)

£12.70 for six issues UK ☐

£14.20 for six issues overseas surface mail ☐

£25.00 for six issues overseas airmail ☐

Subscription Order Form

Please commence my subscription to ZX Computing with the very next issue.

I am enclosing my (delete as necessary) cheque/Postal Order/International Money Order for £: . . . .
(made payable to ASP Ltd)
OR
*Debit my Access/Barclaycard**
*(*delete as necessary)*

We welcome Access

VISA

☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Please use BLOCK CAPITALS and include post codes.

NAME (Mr/Mrs/Miss) . . . .

ADDRESS . . . .

. . . .

. . . . **POSTCODE** . . . .

Signature . . . .

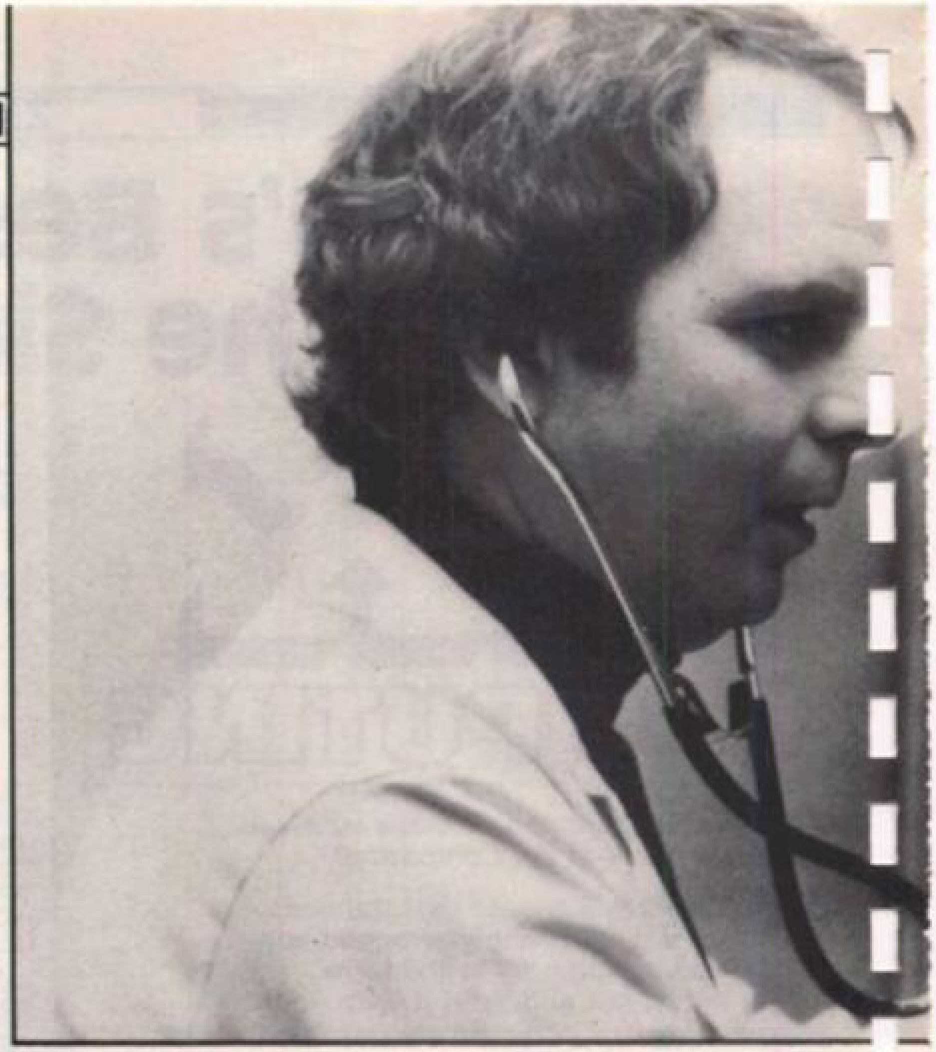
Date . . . .

A superb general purpose program for storing data from Dr R G Shillito in Essex.

As it stands, this program is set up as a general purpose data store for general practitioners! However, for those of you who are not members of the medical fraternity, it is very easy to amend, alter and adjust to suit your own individual needs.

Full instructions are given for the operation of the program as it was written and I am a firm believer in encouraging programmers to adapt programs themselves rather than hand everything to them on a plate.

So here it is, modify it as you will . . .



Data

```
   1 DIM N$(5,5000)
   2 DIM B$(5,32)
   3 DIM t(5)
  10 CLS
  15 PRINT  AT 10,15;"Enter Code
.": PAUSE 0
  20 IF  INKEY$ ≠"S" THEN  GO TO
 55
  25 CLEAR
  30 CLS
  40 NEW
  55 CLS : PRINT ,,"Code Accepte
d.": PAUSE 75: CLS
  60 PRINT  INK 3, AT 0,0;"Do yo
u wish to see instructions  for
use of this file? y/n": PAUSE 0:
 IF  INKEY$ ="y" THEN  GO SUB 80
00
  70 INK 0
  75 GO SUB 9900
  80 PRINT  AT 0,0;"ANY KEY TO C
ONTINUE...": PAUSE 0
 100 CLS : BRIGHT 0: BORDER 7: P
RINT  AT 0,0;"Menu"
 110 PRINT  AT 00,10;"1---Retrie
ve file."
 120 PRINT  AT 02,10;"2---Add to
 file"
 130 PRINT  AT 04,10;"3---Obtain
 hard copy  of a file."
 140 PRINT  AT 06,10;"4---Open a
 new file."
 150 PRINT  AT 08,10;"5---Save f
ile on tape This will save the p
rogram aloneThe notes must be sa
ved by using routine ---7"
 160 PRINT  AT 14,10;"6---To ent
er old notes to this program** T
HIS ROUTINE CLEARS CURRENT NOTES
**"
 170 PRINT  AT 18,10;"7---To sav
e existing notes on tape."
 180 PRINT  AT 20,10;"8---To qui
t program."
 200 INPUT x
 205 IF x=8 THEN  NEW
 210 GO SUB 1000*x
 220 BORDER 7: PAPER 7: INK 0: F
LASH 0: BRIGHT 0
 225 CLS
 230 GO TO 100
1000 CLS
1010 INK 2: PRINT "*Press 'y' wh
en asked 'Scroll?'* ": PAUSE 100
: CLS : INK 0
1015 GO SUB 9900
1020 INPUT "Enter file number.1-
5.";a: CLS
1030 BORDER a
1060 PRINT  AT 0,0;B$(a)
1065 GO SUB 9000
1075 PRINT  AT 6,0;N$(a)( TO t(a
)+96)
1076 INK 2: PRINT  AT 20,0;"To r
eview notes after this print pre
ss 'y','n'to return to menu."
1080 INK 0: PAUSE 0: IF  INKEY$
="n" THEN  GO TO 1090
1085 GO TO 1015
1100 BORDER 7
1500 RETURN
2000 CLS : PRINT  AT 0,0;"This r
outine adds to the notes  at pre
```



store

```
sent on this file. Any key to co
nt.": PAUSE Ø: CLS
2005 GO SUB 9900
2010 INPUT "Enter File No.to be
added to.1-5";a: CLS
2020 BORDER a
2050 GO SUB 9000
2100 GO SUB 4041
2300 RETURN
3000 CLS
3002 GO SUB 9900
3005 INPUT "Enter the file no to
 be printed.1-5",a
3006 CLS : BORDER a
3010 INPUT "Enter 1 for copy of
the whole   file, 2 for a copy o
f a single  page.";,,H
3015 GO SUB 9000
3020 IF H=1 THEN  LPRINT  AT Ø,Ø
;B$(a): LPRINT  AT 2,Ø;N$(a)( TO
 t(a))
3021 IF H=2 THEN  GO TO 3030
3025 GO TO 3100
3030 CLS : PRINT "When the page
you require a copy of appears pr
ess 'z' to obtain your copy. Any
 key to see the next page.Any ke
y to continue.": PAUSE Ø
3033 LET f1=640
3035 LET c=1: LET f2=f1
3040 FOR e=1 TO  INT (t(a)/f1)+1
3050 IF f2>t(a) THEN  LET f2=t(a
)
3060 CLS : PRINT B$(a): PRINT  A
T 2,Ø;N$(a)(c TO f2): PAUSE Ø: I
F  INKEY$ ="z" THEN  LPRINT  AT
Ø,Ø;B$(a): LPRINT  AT 2,Ø;N$(a)(
c TO f2)
3065 LET c=c+f1-1: LET f2=f2+f1
3068 IF f2>t(a) THEN  LET f2=t(a
)
3070 NEXT e
3075 GO TO 3100
3100 CLS : PRINT "Do you want a
copy of another    file? Y/N": PA
USE Ø: IF  INKEY$ ="y" THEN  GO
TO 3000
3500 RETURN
4000 CLS : GO SUB 9900: INPUT "E
nter file No.to be written;1-5."
;a: IF a>5 OR a<1 THEN  GO TO 40
00
4001 CLS : BORDER a: IF  CODE B$
(a)( TO )>32 THEN  PRINT  AT 2,8
; INK 2;"WARNING FILE OCCUPIED."
; AT Ø,Ø;B$(a): PAUSE 300
4002 PRINT  AT Ø,Ø;a;" ";B$(a),
AT 02,Ø;"Press 'y' to erase old
file or  change name/No.; any ke
y to       return to menu. "
4003 PAUSE Ø: IF  INKEY$ ="y" TH
EN  GO TO 4008
4005 GO TO 4100
4008 CLS : INK Ø
4010 INPUT "Enter patients name
and number; then press enter:",B
$(a)
4015 CLS : PRINT  AT Ø,Ø;a;" ";B
$(a); AT 3,Ø;"If you wish to era
se old notes  press'y','n'to ret
urn to menu.": PAUSE Ø: IF  INKE
Y$ ="n" THEN  GO TO 4100
4020 LET t(a)=Ø: LET t=Ø
4025 CLS : PRINT  AT Ø,Ø;B$(a)
4030 PRINT  AT 2,Ø;"Now type in
patient details."
4033 INPUT "When you have finnis
hed press 'enter'",;N$(a)
4040 GO SUB 9000
4043 CLS : PRINT  AT Ø,Ø;B$(a):
PRINT  AT 2,Ø;5000-t(a);" figs.
left in file.": PRINT  AT 4,Ø;N$
(a)( TO t(a)+96); INK 2, AT 20,Ø
;"Do you wish to add to this fil
e?       y/n": PAUSE Ø: IF  INKEY
$ ="n" THEN  GO TO 4090: INK Ø
4046 CLS : PRINT B$(a); AT 2,Ø;N
```

```
$(a)( TO t(a)): INPUT "Add futhe
r info. Then press      enter. ";
M$
4047 LET N$(a)=N$(a)( TO t(a))+M
$
4048 GO SUB 9000
4049 CLS : PRINT  AT 0,0;B$(a):
PRINT  AT 2,0;N$(a)( TO t(a)): I
NPUT ;"Any No. to continue.";k
4050 CLS : INK 2: BRIGHT 1: FLAS
H 1
4051 PRINT  AT 0,6;" IS FILE COM
PLETE? "; AT 1,14;"y/n"; AT 10,4
;5000-t;" figs. left in file ";a
: PAUSE 0: IF  INKEY$ ="y" THEN
 GO TO 4100
4052 FLASH 0: BRIGHT 0: INK 0
4055 CLS
4060 GO TO 4046
4090 INK 0: BORDER 7
4100 RETURN
5000 CLS
5005 FLASH 1: BRIGHT 1: PRINT  I
NK 2, AT 0,1;"HAVE YOU SAVED CUR
RENT NOTES"
5006 INPUT "Enter 1 to cont. any
 no. to      return to menu.";z
5007 FLASH 0: BRIGHT 0: INK 0
5008 IF z <> 1 THEN  GO TO 5500
5009 CLEAR
5010 INPUT "Enter a name for the
 file. eg.  file 1 Up to 10 char
ecters.";,I$
5030 SAVE I$ LINE 1
5035 CLS : PRINT "Wind tape back
 and press any key to verify.":
PAUSE 0
5037 VERIFY I$
5038 PRINT  INK 2;"VERIFIED"
5039 PRINT "Any Key to return to
 menu.": PAUSE 0
5040 GO TO 1
5500 RETURN
6000 CLS
6005 FLASH 1: BRIGHT 1
6010 PRINT  INK 2; AT 0,0;"WARNI
NG THIS ROUTINE CLEARS        EXIST
ING NOTES."
6012 PAUSE 150
6013 INK 0: BRIGHT 0: FLASH 0: C
LS
6015 PRINT  INK 2;"IF YOU WISH T
O SAVE CURRENT     NOTES RETURN
TO MENU BY PRESSING 'y' ANY KEY
TO CONTINUE.": PAUSE 0
6018 IF  INKEY$ ="y" THEN  GO TO
 6500
6025 INK 0
6026 INPUT "Enter the file name.
ie.Patient  initials.",O$
6035 CLS : PRINT "Start tape and
 press any key.": PAUSE 0
6040 LOAD O$ DATA B$(): LOAD O$
DATA N$(): LOAD O$ DATA t()
6100 PRINT "DATA LOADED.ANY KEY
TO RETURN.": PAUSE 0
6500 RETURN
7000 CLS : PRINT "This routine s
aves all five     current notes
on to tape,       without the pr
ogram."
7010 INPUT "Enter a name for the
se notes.eg.'File1-5'";O$
7020 SAVE O$ DATA B$(): SAVE O$
DATA N$(): SAVE O$ DATA t()
7030 CLS : PRINT "Wind tape back
 and hit any key  to verify.": P
AUSE 0
7040 VERIFY O$ DATA B$(): VERIFY
 O$ DATA N$(): VERIFY O$ DATA t(
)
7050 CLS : PRINT  INK 2, AT 0,10
;"NOTES VERIFIED."; AT 3,4;"Any
key to return to menu.": PAUSE 0
7060 INK 0
7500 RETURN
8000 CLS : INK 0: PRINT "The pro
gram is mostly self        explana
tory,however the computer will e
xpect you to respond by  enterin
g a number followed by a single
press of the 'enter' key when a
decision is required.          "

8010 PRINT
8020 PRINT "When recording notes
 it is a     good idea to remove
the 'ear'   connection at the ta
pe as this  results in a better
recording.  When asked to verify
 this       recording the plug s
hould be    inserted again."
8030 PRINT
8040 PRINT "Should the computer
respond with 'R Tape loading err
or' then     press'g'100,this wil
l show on    the screen as'GO TO
100' and     after pressing enter
 will return you to the menu.Pre
ss any key"
8050 PAUSE 0
8060 CLS : PRINT "For reasons of
 data protection  the program ca
nnot be saved with the notes the
se must be saved  and loaded sep
arately,provision for this has b
een made in the   program.It is
suggested that    the  notes be
```

```
recorded on a     separate tape
(s)."
8070 PRINT
8080 PRINT "Press any key.": PAU
SE 0
8090 CLS : PRINT "There are 32 s
paces provided for the patients
name and number, it is therefor
e suggested that the address is
 filed with the   notes. Each on
e of the five     sets of notes
has room for 5,000 characters (i
ncluding blank spaces. "
8100 PRINT
8110 PRINT "Press any key.": PAU
SE 0
8500 RETURN
9000 IF t(a)>1 THEN  LET t=t(a):
 GO TO 9010
9005 FOR t=1 TO 5000
9010 LET x=t-1
9015 IF x=0 THEN  LET x=1
9030 IF  CODE N$(a)(x TO t)=32 T
HEN  GO TO 9050
9040 NEXT t
9050 IF  CODE N$(a)(x+1 TO t+1)=
32 THEN  GO TO 9100
9060 LET t=t+1
9070 GO TO 9010
9100 IF  CODE N$(a)(x+2 TO t+2)=
32 THEN  GO TO 9200
9110 LET t=t+1
9120 GO TO 9010
9200 IF  CODE N$(a)(x+3 TO t+3)=
32 THEN  GO TO 9300
9250 LET t=t+1
9260 GO TO 9010
9300 IF  CODE N$(a)(x+10 TO t+15
)=32 THEN  GO TO 9400
9310 LET t=t+10
9320 GO TO 9010
9400 LET t(a)=t
9410 IF t(a)>1 AND t(a)<32 THEN
 LET t(a)=32
9450 IF t(a)>32 THEN  LET t(a)=
INT ((t(a)/32)+1)*32
9500 RETURN
9900 CLS
9905 FOR d=1 TO 5
9910 BORDER d
9920 PRINT  AT d+4,2;d;" ";B$(d)
9930 IF  CODE B$(d)=32 THEN  PRI
NT  AT d+4,4;"EMPTY FILE."
9940 NEXT d
9950 PRINT  AT 20,0;"THE ABOVE N
OTES ARE CURRENTLY   LOADED INTO
 YOUR COMPUTER."
9999 RETURN
```

NEW FROM

Widgit Software Ltd

"HOW NICE IT IS TO SEE ADVENTURES THAT REALLY MAKE PEOPLE THINK"

"BEST WIDGIT PROGRAM SO FAR"

Castle of Dreams £7·95

A Brain Teasing Exploration for 11 year olds and above

Pit your wits against the wicked Magician Klingsor and save the world from his evil

Adventure Playground £7·95

Games you can play over again.

Two Absorbing Explorations for Juniors

"Crooked Adventure" and "The Queen of Hearts"

You never get the same ones twice

Other titles from WIDGIT: THE HUMPTY DUMPTY MYSTERY £6·25 PATHFINDER £5·95
ALPHABET, COUNTING, SHAPE SORTER & ADDING & SUBTRACTING £5·25 each
Also QUICK THINKING & LOOK SHARP published by MIRRORSOFT

WIDGIT SOFTWARE is available from
48 DURHAM ROAD, LONDON N2 9DT

and from Boots WHSMITH
and your usual computer shop

HISOFT PASCAL DEVPAC

Quality System Software

HISOFT PASCAL 4T

"... I haven't seen any other compiler that could match Hisoft's Pascal" ... *Using Spectrum Micro — Autumn 1983.*

"This is a very impressive product ... of benefit to any Spectrum programmer" ... *David Bolton ZX COMPUTING Aug/Sept 1983.*

Just two comments from full length reviews of our powerful and virtually full implementation of Standard Pascal. The advantages of using the Pascal language are many: fast, self-documenting and truly structured programming — many schools and universities insist on a good knowledge and understanding of Pascal for their Computer Studies courses.

Now you can learn and use Pascal on your home micro: Hisoft Pascal is available for a wide range of home computers, including the 48K Spectrum. Typically, compiled programs run 40 times faster than their interpreted BASIC equivalent — and sometimes up to 1,000 times faster!

Hisoft Pascal supports all the Standard Pascal data structures and statements: INTEGERs, REALs, CHARacters, ARRAYs, SETs, RECORDs, POINTERs, Enumerated types, FOR ... DO, REPEAT ... UNTIL, WHILE ... DO, CASE ... OF and many pre-defined Procedures and Functions (e.g. SQRT, TAN, SIN, COS, INLINE, PEEK, POKE etc. etc.) — it is **not** a Tiny Pascal but a powerful educational and development tool which allows the user to develop true high-level language skills, while attaining execution speed close to that of machine-code.

Hisoft Pascal for the ZX Spectrum now comes complete with ZX Microdrive support (anything you can do with tape you can do with Microdrive) and a Turtle Graphics package which allows easy creation of complex graphic programs.

HISOFT DEVPAC 3

"... DEVPAC is most highly recommended. The documentation is first class" ... *Your Computer — May 1983.*

"... it is impossible to recommend any other development package for the Spectrum over DEVPAC." ... *Adam Denning — ECM April 1984.*

Need we say more? DEVPAC 3 is a very powerful assembler, editor and disassembler/debugger. Forget about buying lots of separate packages for your machine-code development — it's all in DEVPAC 3. A full Z80 assembler with conditional assembly, all standard directives, full arithmetic (*, /, +, −, OR, XOR, AND, MOD), many assembler commands for controlling object code generation, listing, printing etc., full line editor with advanced facilities like search/replace string, 'include' facility that lets you assemble from tape (or Microdrive) without taking up memory for your source text **AND** a superb disassembler/debugger with 'front panel' display of Z80 registers and flags with commands like search for string of bytes, copy memory, disassemble memory (to an assembler textfile, if you wish), list memory in ASCII, set multiple breakpoints, and, above all, single-step through machine code programs, one instruction at a time, **EVEN IN ROM!**

DEVPAC comes complete with a 45 page user manual.

On the ZX Spectrum, DEVPAC fully supports the ZX Microdrive, allowing assembly from microdrive, saving of text to and from Microdrive and saving of object code to Microdrive — DEVPAC still fully supports tape as well.

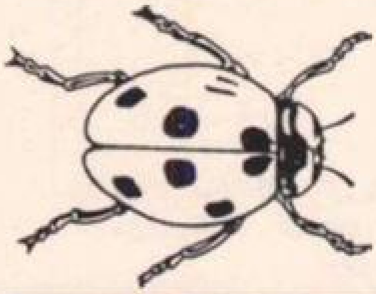
Prices:

Hisoft Pascal 4T — **(ZX SPECTRUM) £25 inclusive**
(NewBrain, SHARP MZ700 etc) £35 plus VAT
Hisoft Pascal 4D, many disk formats £46 inc.

Hisoft DEVPAC 3 — **(ZX SPECTRUM) £14 inclusive**
(NewBrain) £25 inclusive

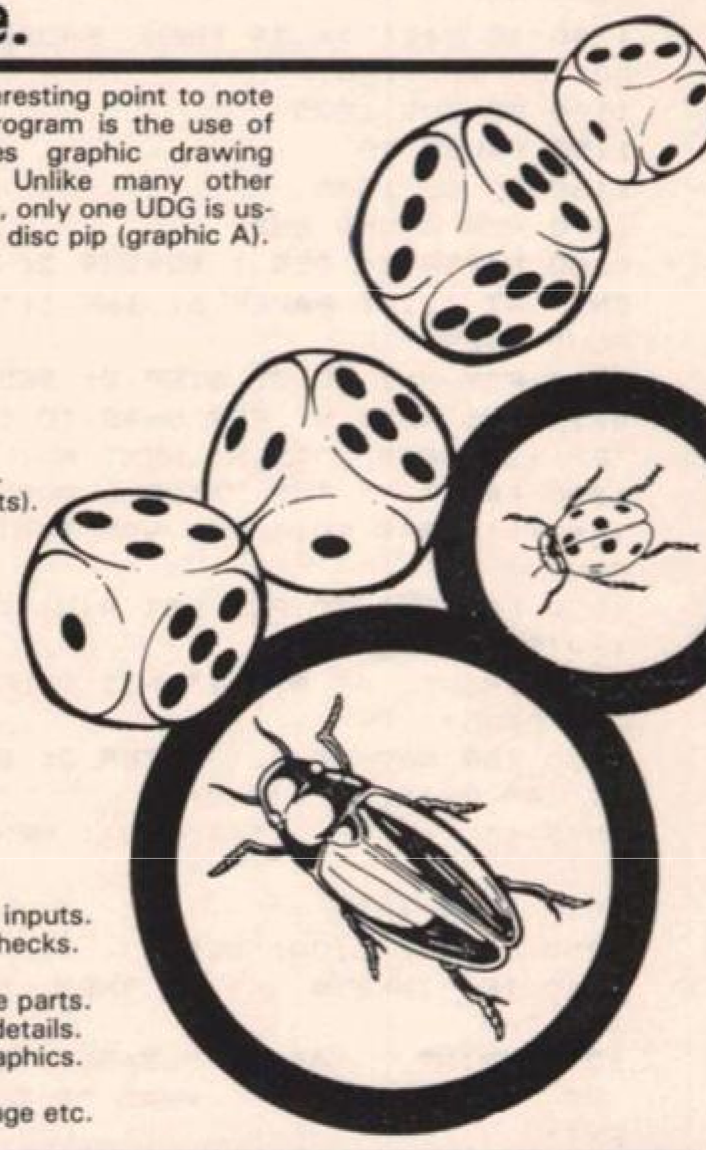
HISOFT
13 Gooseacre, Cheddington
Leighton Buzzard, Beds. LU7 0SR
Tel: (0296) 668995

*and others



Beetle

There are no bugs in this game! Up to four players can take part in Colin Gooch's realistic dice game.



Many traditional games have made their way from boards, cards, pen and paper into the computer screen, some successfully and some not. This is one of the sucessful transplants.

For those of you who have missed the delights of this game in its original incarnation, the idea is to throw a dice and build up a "beetle" according to the numbers thrown.

The computer displays the cards and the stages of the Beetle as it is drawn and a clever routine is used to simulate dice throwing. Instead of just saying "You threw 3", or whatever (a very suspicious format to non-computer-expert players) a dice is displayed which "rolls' until the player removes his finger from the key. This gives a realistic feel to the game.

When less than four players take part then the computer joins in. It is a mean opponent, and I can't spot any sneaky programming which gives it an advantage!

You must have a 6 to start and this draws the body, 5 draws the head and other numbers provide antenna, legs, eyes, etc. As you will gather, this is a game where luck is the important factor, but it makes a nice relaxing change from saving the world from aliens, or wrestling with some logical computer generated problem.

An interesting point to note in this program is the use of the hi-res graphic drawing routines. Unlike many other programs, only one UDG is used — the disc pip (graphic A).

Variables used

A$,B$,C$	Dice spots.
D$,E$	Dice frame.
P$(N)	Players' names.
S$	Spectrum control, Play or Not.
N$,Z$	Temporary variables (line inputs).
A(X)	Accumulated totals.
B(X,Y)	Beetles.
Count	Control variable in dice.
D	Dice value.
FF	Dice control variable.
N	FOR/NEXT loops.
PL	Number of players.
T	Turn number.
V(X)	Value of score.
X,Y,P,I	Variables to set up title page.

The lines

1000-1020	Introduction, initialisation and inputs.
1030-1060	Main control loop, including checks.
1180-1320	Endgame routines.
1340-1600	Subroutines to print out beetle parts.
1610-1810	Start — get inputs on player details.
1820-1960	Initialise variables and dice graphics.
1970-2090	Dice throwing routine.
2100	Miscellaneous drawing title page etc.

```
 500 REM #########################
################################
+++++     B E E T L E    ++++++++
+++++ @ C.N.GOOCH 1983   ++++++++
################################
################################
 600 REM >>>>>>>>>>>>>>>>>>>>>>>>
>>>>>Graphic in lines 1860/1950
is a "GRAPHICS" 'A'<<<<<<<<<<<<<
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
1000 PAPER 3: BORDER 1: CLS : PR
INT  AT 10,4; FLASH 1;" STOP THE
 TAPE ": GO SUB 2100
1010 GO SUB 2110: GO SUB 1820
1020 GO SUB 1620: GO SUB 1500
1030 REM MAIN LOOP
1040 PRINT  AT 14,0:"
                        "; FLASH 1;
INK 0; PAPER 2+T; AT 14,T*8-6;"
GO "
1050 GO SUB 1970
1060 IF D=1 AND B(T,5) AND B(T,1
)<2 THEN  LET B(T,1)=B(T,1)+1: L
ET V(T)=V(T)+1: GO SUB 1390+((B(
T,1)=2)*10)
1070 IF D=2 AND B(T,5) AND B(T,2
)<2 THEN  LET B(T,2)=B(T,2)+1: L
ET V(T)=V(T)+2: GO SUB 1440+((B(
T.2)=2)*10)
1080 IF D=3 AND B(T,6) AND B(T,3
)<4 THEN  LET B(T,3)=B(T,3)+1: L
ET V(T)=V(T)+3: GO SUB 1470+((B(
```

```
T,3)=1 OR B(T,3)=3)*10)
1090 IF D=4 AND B(T,6) AND B(T,4
)=0 THEN  LET B(T,4)=1: LET V(T)
=V(T)+4: GO SUB 1410
1100 IF D=5 AND B(T,6) AND B(T,5
)=0 THEN  LET B(T,5)=1: LET V(T)
=V(T)+5: GO SUB 1370
1110 IF D=6 AND B(T,6)=0 THEN  L
ET B(T,6)=1: LET V(T)=V(T)+6: GO
 SUB 1350
1120 PRINT  AT 12,T*8-7;"Sc ";V(
T)
1130 IF V(T) >=.33 THEN  PAUSE 1
00: GO TO 1180
1140 GO SUB 1800
1150 PAUSE 75
1160 GO TO 1040
1170 REM ROUND END
1180 PAPER 1: CLS : BORDER 2: PR
INT  AT 1,10; PAPER 6; INK 1;" R
OUND OVER "
1190 FOR N=1 TO 32 STEP 2: BEEP
N/100,N: NEXT N: FOR N=40 TO 1 S
TEP -1: BEEP .005,N: NEXT N
1200 PRINT '' TAB 7;"WELL DONE "
;P$(T);' TAB 10;" YOU WON THAT R
OUND "
1210 FOR N=1 TO PL: LET A(N)=A(N
)+V(N): NEXT N
1220 PRINT  AT 8,10;"THIS ROUND
   TOTAL"
1230 FOR N=2 TO PL*2 STEP 2: PRI
NT  AT N+10,1;P$(N/2);"       ";V
(N/2);"      ";A(N/2): NEXT
N
1240 BEEP 1,20
1250 GO SUB 2100: CLS
1260 IF  INKEY$ <> "" THEN  GO
TO 1260
1270 PRINT ' TAB 2;"PLEASE PRESS
 KEY TO TELL ME","  WHAT TO DO N
EXT"
1280 PRINT '"  (1)PLAY ANOTHER R
OUND WITH","   THE SAME PEOPLE
"," (2)PLAY A NEW GAME WITH ","
   DIFFERENT PEOPLE","  (3)FIN
ISH PLAYING BEETLE"
1290 IF  INKEY$ ="1" THEN  GO SU
B 1500: DIM B(4,6): DIM V(4): GO
 TO 1040
1300 IF  INKEY$ ="2" THEN  GO TO
 1020
1310 IF  INKEY$ ="3" THEN  GO SU
B 2110: FOR N=8 TO 14: PRINT  AT
 N,5; PAPER 7;"
   ": NEXT N: PRINT  PAPER 6; IN
K 2; BRIGHT 1; AT 10,6;"THANKS F
OR PLAYING"; AT 11,6;"START TAPE
 TO LOAD"; AT 12,6;" THE NEXT G
AME   ": LOAD ""
1320 GO TO 1290
1330 REM BEETLES
1340 REM BODY
1350 PLOT T*64-44,119: DRAW 24,0
,- PI : DRAW 0,-8: DRAW -24,0,-
PI : DRAW 0,8: RETURN
1360 REM HEAD
1370 CIRCLE T*64-32,138,8: RETUR
N
1380 REM EYES
1390 CIRCLE T*64-35,140,1: RETUR
N
1400 CIRCLE T*64-29,140,1: RETUR
N
1410 REM TAIL
1420 PLOT T*64-32,98: DRAW 0,-8,
 PI : DRAW -6,0,- PI : RETURN
1430 REM FEELERS
1440 PLOT T*64-36,144: DRAW -5,3
: DRAW -2,-6,.3* PI : RETURN
1450 PLOT T*64-26,144: DRAW 5,3:
 DRAW 2,-6,-.3* PI : RETURN
1460 REM LH LEG
1470 PLOT T*64-44,119-8*(B(T,3)=
4): DRAW -5,0: DRAW -6,-6,.5* PI
 : RETURN
1480 REM RH LEG
1490 PLOT T*64-20,119-8*(B(T,3)=
3): DRAW 5,0: DRAW 6,-6,-.5* PI
: RETURN
1500 REM set screen
1510 BORDER 0: PAPER 0: CLS : IN
K 7
1520 PRINT AT 0,10; PAPER 6; IN
K 2;" BEETLE "
1530 PLOT 0,167: DRAW 255,0: PLO
T 0,64: DRAW 255,0
1540 PLOT 0,166: DRAW 0,-101
1550 PLOT 63,166: DRAW 0,-101
1560 PLOT 127,166: DRAW 0,-101
1570 PLOT 191,166: DRAW 0,-101
1580 PLOT 255,166: DRAW 0,-101
1590 FOR N=8 TO 8*PL STEP 8: PRI
NT  AT 2,N-7; INK 0; PAPER N/8+2
;P$(N/8): NEXT N
1600 RETURN
1610 REM START
1620 CLS : BORDER 0: PAPER 0: IN
K 7
1630 DIM P$(4,6)
1640 CLS : PRINT ; PAPER 6; INK
0; AT 2,2;" HOW MANY PEOPLE  TO
PLAY ?"; PAPER 0;
1650 INPUT  LINE N$: IF  CODE N$
<49 OR  CODE N$>57 THEN  PRINT
PAPER 2; INK 7;''"ENTER A NUMBER
 PLEASE ": GO TO 1650
1660 LET PL= VAL N$: IF PL<1 OR
```

```
PL>4 THEN  PRINT  PAPER 2;'" I A
M SORRY BUT THERE ARE ONLY  FOUR
 CARDS,  PLEASE ENTER AGAIN ": G
O TO 1650
1670 IF PL=1 THEN  PRINT  PAPER
6: INK 0:''" YOU WILL WANT ME TO
 PLAY THEN ": LET S$="YES": LET
PL=2: LET P$(2)="'PUTER": PAUSE
250: GO TO 1730
1680 IF PL=4 THEN  LET S$="NO":
GO TO 1730
1690 CLS : IF PL<4 AND PL>1 THEN
  PRINT  PAPER 6; INK 0;''"DO YO
U WANT ME TO PLAY ?": INPUT  LIN
E S$
1700 IF S$(1)="N" OR S$(1)="n" T
HEN  PRINT  PAPER 4; INK 0;''"O.
K. I'LL JUST WATCH": GO TO 1730
1710 IF S$(1)="Y" OR S$(1)="y" T
HEN  PRINT  PAPER 6; INK 0''" GO
OD . I LIKE PLAYING ": LET PL=PL
+1: LET P$(PL)="'PUTER": LET S$=
"YES": GO TO 1730
1720 PRINT  PAPER 2;''"SORRY.I D
ID NOT   UNDERSTAND "'"PLEASE AN
SWER ""YES"" OR ""NO""": INPUT S
$: GO TO 1700
1730 PRINT ''"  PLEASE ENTER THE
 PLAYERS          INITIALS OR NAMES
 ...NOT MORE  THAN SIX LETTERS E
ACH"
1740 FOR N=1 TO PL-(S$(1)="Y")
1750 PRINT '"PLAYER ";N : INPUT
 LINE P$(N): IF P$(N,1)=" " THEN
  GO TO 1750
1760 PRINT P$(N): NEXT N
1770 IF S$(1)="Y" THEN  PRINT '"
PLAYER ";PL: PRINT "'PUTER"
1780 GO SUB 2100
1790 LET T=1+ INT ( RND *4)
1800 LET T=T+1: IF T >= PL+1 THE
N  LET T=1
1810 RETURN
1820 REM INIT
1830 POKE 23658,8: RESTORE 2150:
 DIM B(4,6): DIM A(4): DIM V(4):
 FOR N=0 TO 7: READ U: POKE  USR
 "A"+N,U: NEXT N
1840 REM #### DICE ####
1850 DIM A$(6,3): DIM B$(6,3): D
IM C$(6,3)
1860 LET A$(1)="   "
1870 LET B$(1)=" A "
1880 LET C$(1)=A$(1)
1890 LET A$(2)="A  "
1900 LET B$(2)=A$(1)
1910 LET C$(2)="  A"
1920 LET A$(3)=A$(2): LET B$(3)=
B$(1): LET C$(3)=C$(2)
1930 LET A$(4)="A A": LET B$(4)=
A$(1): LET C$(4)=A$(4)
1940 LET A$(5)=A$(4): LET B$(5)=
B$(1): LET C$(5)=A$(4)
1950 LET A$(6)="AAA": LET B$(6)=
A$(1): LET C$(6)=A$(6)
1960 LET D$= CHR$ 139+ CHR$ 131+
 CHR$ 131+ CHR$ 131+ CHR$ 135: L
ET E$= CHR$ 142+"▬▬▬"+ CHR$ 141:
 RETURN
1970 REM ###PRINT DICE ###
1980 LET FF=0: LET COUNT=0
1990 FOR N=15 TO 19: PRINT  AT N
,5;"                         ": NE
XT N: PAUSE 90
2000 LET D=1+ INT ( RND *6): PRI
NT  AT 15,5;D$: AT 19,5;E$
2010 FOR N=16 TO 18: PRINT  AT N
,5;"▌": AT N,9; CHR$ 133: NEXT N
2020 IF S$="YES" AND T=PL THEN
PRINT  AT 17,12:"MY GO": LET FF=
1: GO TO 2040
2030 PRINT  AT 16,12;"PRESS ANY
KEY TO": AT 17,12;" ""THROW"" DI
CE ": AT 18,11;" RELEASE TO STOP
": IF  INKEY$ ="" THEN  GO TO 20
30
2040 IF D=7 THEN  LET D=1
2050 PRINT  AT 16,6;A$(D); AT 17
,6;B$(D); AT 18,6;C$(D)
2060 IF FF THEN  LET COUNT=COUNT
+1: IF COUNT>50 AND  RND >.5 THE
N  LET FF=0: PAUSE 25: RETURN
2070 IF FF THEN  LET D=D+1: GO T
O 2040
2080 IF  INKEY$ ="" THEN  PAUSE
25: RETURN
2090 LET D=D+1: GO TO 2040
2100 INPUT "PRESS ENTER TO CONTI
NUE"; LINE Z$: RETURN
2110 CLS : PAPER 0: BORDER 0: CL
S
2120 FOR N=1 TO 30: LET X=1+ INT
 ( RND *23): LET Y=1+ INT ( RND
*20): LET P=1+ INT ( RND *7): LE
T I= INT ( RND *7)
2130 BORDER P: BEEP I/100,P: PRI
NT  AT Y,X; INK I; PAPER P;"BEET
LE"
2140 NEXT N: GO SUB 2100: RETURN
2150 DATA 0,0, BIN 00111100, BIN
 00111100, BIN 00111100, BIN 001
11100,0,0
9995 SAVE "BEETLE" LINE 1000:
 CLS : PRINT  AT 5,2; FLASH 1;"O
 K SET RECORDER "; AT 6,2;"FOR V
ERIFY"; AT 7,2;"START THE TAPE":
 VERIFY "BEETLE": CLS : PRINT  A
T 10,10:"O K"
```

Spectrum ROM routines

A guided tour of some of the Spectrum ROM by Adrian Marsh of Clanfield.

The Spectrum ROM holds many secrets. The most interesting are the monitor routines and in this article I shall outline a few of these routines, and explain how they may be used.

On screen printing

There are a number of ways of transferring information to the screen:

1 Hex address: 0B24-0BDA. This is the PRINT-ANY character subroutine. On entry the HL register pair holds the pixel address of where the character is to be printed; the BC register pair holds the current line and column values and the A register holds the character code.

This method is complicated so it may be easier to use the second method.

2 This subroutine is contained within another, therefore the whole routine begins at 0D68 and ends at 0EAB. At the present the reader is interested in location 0DD9. This sets the printing locations to BC:

i) Load the BC register pair with the appropriate values.
ii) Call CL-SET and Call 0DD9 which enters required values.

BC is equivalent for a position AT a,b; the B register holds the line (hex) 18-a and the C register holds the column (hex) 21-b. Therefore, a routine to produce an equivalent to PRINT At 11,15; would be:

```
01 13 0D    LD BC, 0D13
CD D9 0D    CALL 0DD9
3E ??       LD A, (character)
D7          RST 0010
C9          RET
```

3 The routine PR-STRING can be used to print any string. The monitor routine consists of:

PR-STRING address 203C

```
(label) PR-STRING
LD A,B,
OR C
DEC BC
RET Z
LD A,(DE)
INC DE
RST 00010
JR PR-STRING
```

Any string of characters can therefore be printed by:

i) Loading the start address into DE
ii) Loading the length of the address into BC
iii) Call the PR-STRING at address 203C



Clearing the screen

The B register holds a value in the range of (hex) 01-18. Therefore, (hex) 18 would clear the whole screen.

The CL-LINE routine begins at 0E44 and is very short:

```
06 18       LD B, 18
CD 44 0E    CALL CL-LINE
C9          RET
```

This will clear the entire screen whereas if B were loaded with ▶

17, all but the top line would be cleared.

Sound

There are two routines in the ROM for producing sound, the BLEEPER and the BEEP.

1 The hex address for the BLEEPER is 03B5-03F7. On entry the HL register pair hold the pitch and the DE register pair holds the duration. The duration value has to be increased as the pitch value is decreased. The pitch for middle C is 0666 and the duration for a second is 0105 so a routine for BEEP 1,0 would be:

```
11 05 01    LD DE, 0105
21 66 06    LD HL, 0666
CD B5 03    CALL BLEEPER
C9          RET
```

1 The hex address for the BEEP routine is 03F8-046D. This routine uses the calculator to change the duration and pitch into appropriate values for the DE and HL register pairs.

Using the keyboard

The main routine is the KEY-SCAN.

1 Hex address for the KEY-SCAN is 028E-02BE. On leaving the routine, DE is returned with a key value. The zero flag is reset if more than one key is pressed at the same time. The D register indicates which shift keys are being pressed and the E register contains the key number (hex) 00-27.
2 At location 02BF-03B4 are found the KEYBOARD subroutines. These handle the repeat facility and decode the key-value to give the required character code. If the code is accepted then it is placed in the system variable LAST-K and bit 5 of the FLAGS is set.
3 At location 10A8-111C there is the KEYBOARD-INPUT routine. This routine copies the value from LAST-K and depends on bit 5 of the FLAGS. It then returns with the carry flag set or reset if the code is printable.
4 The subroutine at location 15D4-1651 literally has the effect of a PAUSE 0 or 'waiting for a key to be pressed'. So, a pause 0 in machine code would consist of:

```
CD D4 15 CALL 15D4
```

Loading and saving

The whole routine starts at 04C2 and ends at 09F3.

1 SAVING This subroutine begins at 04C2-053E and passes the DE bytes by starting at the (IX) location and continuing to the cassette recorder with the initial marker byte and parity byte. An example of this is:

i) The amount of bytes to save is 255 (FFh)
ii) The block is stored at location 25000 (61A8h)

Therefore:

```
3E FF           LD, A, 255
DD 21 A861      LD IX, + START
110001          LD DE, + COUNT
C9              RET
```

This routine saves 255 bytes starting at 25000. However, these bytes are saved without a header, and can only be loaded if the count is known.

2 LOADING This subroutine begins at 0556-0604 and loads the DE bytes and the IX register pair points to the first location. When loading the carry flag must be set, but if it is reset then VERIFY can be used. Therefore:

```
(37)            (SCF loading only)

3E FF           LD A, FF
0D 21 A861      LD IX, START

11 00 01        LD DE, COUNT
CD 56 05        LOAD ROUTINE
C9              RET
```

This routine loads 255 bytes into 25000.

I shall now continue to shed some light on the RST commands or restart commands.

1 RST 0 (0000-0007h). This does a number of things in this order:

i) Disable the maskable interrupt
ii) Clears the A register
iii) Loads the DE register pair with 65535 (FF-FFh)
iv) Jumps forward to 11CB

At 11Cb are the initialisation routines (equivalent to NEW).

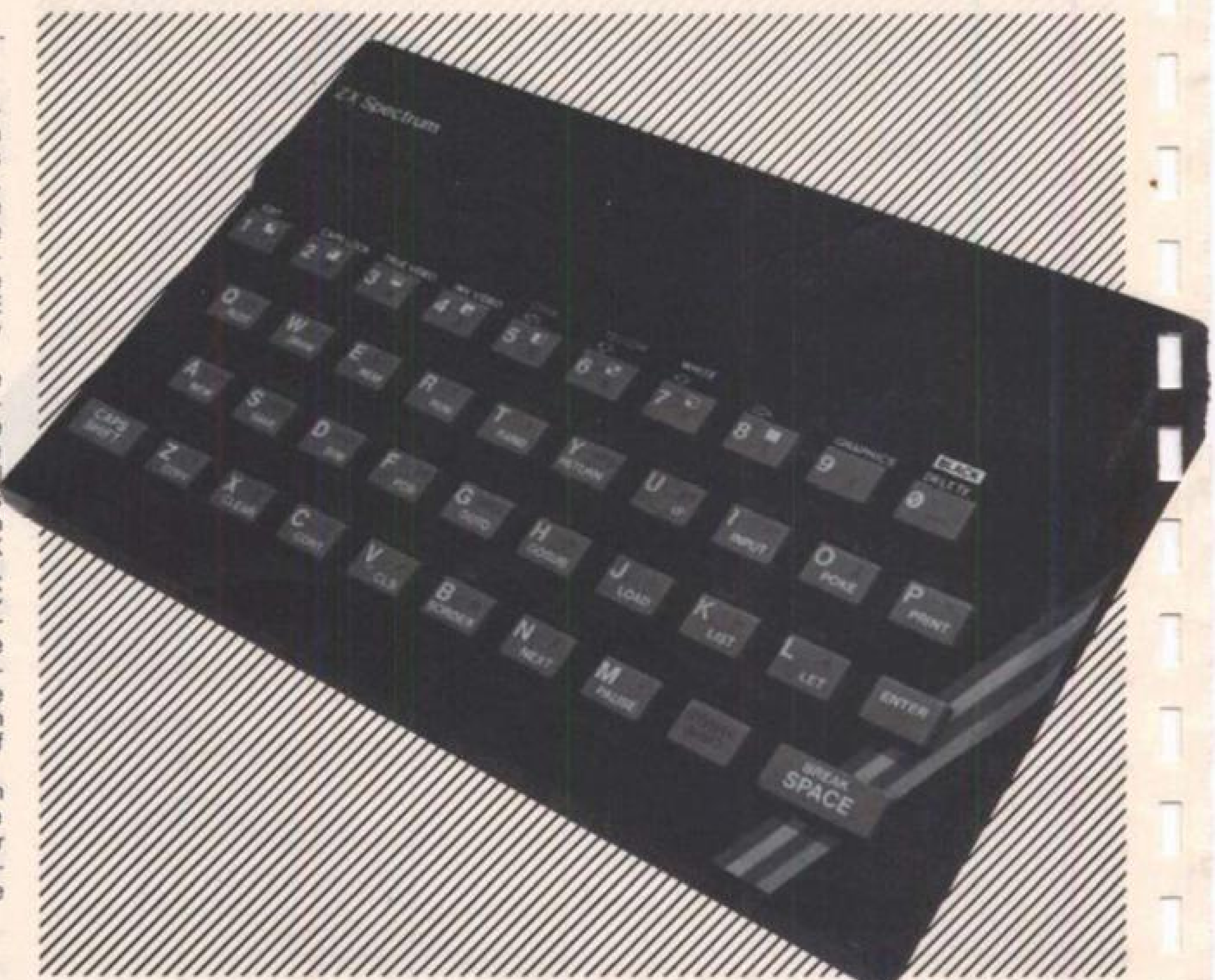
2 RST 8 (0008-000Fh). These are the error routines. There are two outcomes of this:

i) The stack will be cleared.
ii) The appropriate report is given.

3 RST 10 (0010-0012). This is the PRINT A character routine. A jump forward to location 15F2 is made.
4 RST 18 and RST 20 (0018-0024h). This fetches the current character pointed to by the CH-ADD routine.
5 RST 28 (0028-0029h). This jumps forward to location 335B which is the first address of the calculator.
6 RST 30 (0030-0037h). This is the BC-SPACES and jumps forward to 16 9E to make a space in the work space.

A useful routine at 11DA is the RAM-CHECK routine. All locations from RAMTOP to 4000 are tested. The HL register pair holds the address of the last location of memory available.

By now the reader should have a good idea of the monitor programs in the ROM. Although I have only outlined a handful, they are sure to come in useful in writing new machine code routines.



RICHARD SHEPHERD SOFTWARE

A HOME BUDGETING & BANKING SYSTEM THAT REALLY MAKES YOUR 48K SPECTRUM WORK!



A professional style multi function Cash Controller program that can load and make an entry in just 90 SECONDS . . . thanks to its ZX MICRODRIVE COMPATIBILITY. If you're not ready to upgrade to Microdrive yet, this easy to use cassette program can handle up to 400 transactions with your Spectrum, almost filling the 48K memory of your computer.



BANK ACCOUNT

- Statements on demand from any date to date.
- Holds up to 400 transactions.
- Standing order facility.
- All transactions can be automatically coded to allocate up to 16 budget headings.
- Search facility for any single item by description or amount.

HOME BUDGETING

- 16 budget headings, e.g. Gas, Rates, Car, Tax - you can choose the headings
- Highlights under/over spending
- Full budget expenditure breakdown
- Budget "re-think" facility

LOAN/MORTGAGE CALCULATOR

All you ever wanted to know but didn't like to ask! Calculates:

- **Interest rates**
- **Payback periods**
- **Capital sum**
- **Period of loan**

REMEMBER THIS CASSETTE CAN ALSO BE USED WITH ZX MICRODRIVE

- **PRINTOUT FACILITY USING ZX PRINTER**
- **FULL SECURITY WITH PERSONAL PASSWORD**
- **FULL SAVE PROGRAM ONTO CASSETTE OR MICRODRIVE**

Available from good computer stores or direct by mail order... **at a price that won't upset the Bank Manager!**



FIRST FOR MICRODRIVE

Dealer enquiries welcome. Generous discounts.

PRIORITY MAIL ORDER

Please send me ______ CASH CONTROLLERS at £9.95 each including post & packing (overseas orders add £1.50).

Name ______

Address ______

ZX/6/84

I enclose cash/cheque/P.O. payable to Richard Shepherd Software.

Or credit my Access/VISA Card Number ______

Signature ______

MICRODRIVE CARTRIDGE NOT INCLUDED

Credit Card Hotline (06286) 63531

RICHARD SHEPHERD SOFTWARE

ELM HOUSE, 23-25 ELMSHOTT LANE, CIPPENHAM, SLOUGH, BERKS. TEL. (06286) 63531

Readers' reviews

Once again we throw caution to the wind and air your opinions on the software currently available for the ZX machines.

Bridge Player
CP Software
S Datoo

If you are a keen bridge player and have sometimes been stuck at making up a foursome, then I can recommend CP Software's 'Bridge Player' for the 48K Spectrum. Unlike some other bridge games software on the market which basically consist of bridge tutors with pre-determined hands and play, Bridge Player allows you to actually bid for your contract, as well as play each hand as you see fit.

On loading the program, you are greeted by a very cleverly designed title, making use of the four suits. You are then asked to wait while the computer shuffles, deals and sorts the cards. Following that you are given the option of either playing the game properly, so that only your hand is revealed and the bidding for the contracting commences, or alternatively you may 'cheat' and see all four hands and then decide on the contract.

The play then begins with the lead from East or West, depending on who is declarer, after which dummy's hand goes down. You are always South.

As far as the bidding goes, the computer uses the standard ACOL system and also understands and accepts BLACKWOOD and STAYMAN conventions. It will not however, generate bids using these conventions, but will only respond to them.

Apart from the 'table' in the centre of the screen and declarer's and dummy's hands, the display shows the hand number, the dealer, whether any pair is vulnerable and the contract. The number of tricks won by each pair is shown at the bottom left of the screen, and a flashing indicator on the bottom right shows the winner of the last trick.

The faint-hearted have the option of quitting if they find they have entered into an impossible contract. This is done by entering the word "next" instead of a card, and the Spectrum goes on to the next hand. If you play the wrong card when you should follow the suit being played, the Spectrum will forgive your "faux-pas" and revoke the trick without penalty.

When the hand has been played, the display changes to a score card and here a nice little touch has been added. If the contract has been defeated you will be commiserated with a few bars of "The Funeral March" but if you have won the rubber, the Spectrum will play some of Cliff Richard's "Congratulations". You will also be told how much you have won if each point is worth one pound!

The only criticism I have of the program is that you always have to play the contract as declarer and dummy and East/West are always the defenders. You may certainly, during the bidding stage, leave East/West in a contract but you will be informed that you and your partner have the majority of points between you. You will then be asked if you wish to rebid or play the next hand, without the computer having to play as declarer/dummy as well as your partner. It would improve the program immensely if the opposition could also play as declarers.

Hunter Killer
Protek
L Tout

Imagine you're the commander of a British "S" type submarine on an important mission off the coast of Germany and that you have to hunt and destroy enemy submarines. Well this is exactly what you do, if like me, you have bought Protek's submarine simulation program, Hunter-Killer.

On loading an impressive picture showing a submarine launching a torpedo is drawn. You are then asked if you would like a quick practice shot at a submarine. Because the program is a simulation, there is a lot of cunning strategy involved.



In all there are 18 keys to master, which I found did not take long although finding the enemy took a little longer owing to some confusion over bearings, but they soon disappeared.

Your craft has two engines; one diesel and one electric. On the surface you should use the diesel engine but because this needs air to run, the electric engine is used under water. However, if the submarine stays down too long you won't be left with enough power to surface and recharge, so an eye must be kept on the battery charge indicator. The speeds for the diesel and electric engines are 16 and 9 knots respectively, so it's better to use the diesel engines to reach the enemy more quickly.

To make the submarine dive, the ballast tanks must be flooded using key N, and the hydroplane angle altered, using key 6. To stop diving and remain at a constant depth, the hydroplanes must be horizontal and the ballast tanks emptied. Surfacing can be accomplished by blowing the tanks (key B) and raising the hydro-planes (key 7).

Every submarine has to have a periscope and this one is no exception. Key p raises and lowers it and key V lets you look through it.

The water level is shown (which corresponds to your depth) and the enemy, if in visual range. Along the bottom of the screen are shown the periscope angle, your heading (so many degrees from North), the distance to the enemy and the torpedo count. Key O turns the periscope clockwise and key I, anticlockwise.

To help with navigation there are three types of radar; long range (22 miles), short range (4 miles) and an echo sounder which gives the familiar, hollow echo sound, when the submarine is below 15 feet in depth.

There are three high resolution screens shown. All the gauges and dials are shown in the control room which is scrolled sideways on pressing key C, to take you into the Chart Room. The other screen is your view from the periscope. In the Chart Room a map shows the area which you are in and part of the German Coast. Also shown are a tide indicator which tells you its strength and direction, the enemy's last reported position, your position and places where there are mines lurking.

Once the enemy is in range (three miles) and is dead ahead, pressing T will prime the torpedos and fire the first one.

Second and subsequent torpedos are then fired using key F. Their speed is 45 knots and when you peer through the periscope, a bubble track can be seen heading towards the enemy submarine (or not as the case may be). If your aim is true, the submarine can be seen exploding.

Since you are in enemy waters you also encounter aerial attacks. Through the periscope a plane can be seen and heard and then a klaxon sounds. If this happens you have roughly two minutes to dive below 30 feet before you see the screen shake due to the shock waves from the depth charge dropped by the plane. If you fail however, there is the sound of rushing water and which ever room you are in slowly fills up with water. The sound effects are brilliant.

Hunter-Killer has five levels of difficulty and a manual is supplied with the program. A unique feature of this game is to let you play against a friend, each hunting the other but alas, this can only be done if you have a Microdrive to link both Spectrums. One other drawback is that it takes a long time to reach the enemy but this is because it's a simulation program and Protek have based your submarine on a British "S" type craft with true speeds of 9 and 16 knots, nevertheless, this is a very small snag in an excellent program.

I have only given you the bare facts of how to play the game, but there are many more factors which contribute to the need of intricate strategy. This in itself gives the game millions of variations.

Hunter-Killer is for the 48K Spectrum and is priced at £7.95.

The Alchemist
Imagine
J Richards

The Alchemist is not just another boring adventure game that takes hours of typing and frustration, it is a truly excellent arcade-type graphic adventure, similar in style to Hunchback.

The 48K program LOADed first time, taking about five minutes. There's a SCREEN$ and once LOADed you are greeted with quite a catchy tune, very well matched to the program which is set in the distant past. There's a menu with a good selection of keyboard or joystick options — a joystick certainly makes the game easier to master but using the keyboard is not so bad. There's a wide range of key combinations available and it is quite easy to work out a pattern that suits you.

After making your selection, off you go! The idea of the game is to find four pieces of the 'Spell of Destruction', find the Evil Warlock and cast the spell. You are the Alchemist, with a bit of magic of your own and you are on this great mission to destroy the Warlock. The story-line sounds familiar but the program is well written. Of course, it isn't an easy game: there are 16 rooms, which sounds disappointingly few, but each room takes up two screens and there's plenty to go at. As you move through a room, the background scrolls effectively and on going through a doorway the screen changes to put you in another room.

As the Alchemist you are a very stately figure on the screen. One of your tricks is the ability to transform yourself into an eagle (at the press of a button). This is done well in graphics and as an eagle you can fly to areas the Alchemist could never reach in human form. Learning to fly takes practice but you flap your wings nicely and automatically lower landing gear when necessary!

You have a permanent on-screen record of your vital statistics, namely Spell energy and Stamina, along with a record of which object and/or spell you are carrying. You also have an hour glass and a space ready to deposit any of the parts of the 'Spell of Destruction' you have found. The hour glass tells you how much time you have left in a room before the Warlock locates you and starts sapping at your stamina. This is very dangerous and unless you evacuate the room quickly, will probably be fatal!

You have to keep up your strength by eating food which is scattered around. Spell energy is constantly replenished but is used up every time you cast a spell. Spells and other useful objects are scattered about the lair but are invariably guarded by nasties such as ghosties, skulls or other strange creatures. Combat is almost inevitable, and this saps at your strength.

The game is well thought out and needs dexterity as well as brain power. Although I completed the adventure within three hours, I still find it very enjoyable indeed. The colour and graphics are excellent, particularly because the objects are large and easy to identify. Sound is not used a great deal, but is effective.

Any complaints? Only two. The first is that it is far too easy to accidentally abort the game by pressing "1" in the excitement; the second is that there is no 'hold' feature. I expect this would make the game a bit too easy.

If you're after an adventure that you have a fair chance of completing, without being bored in the slightest when you *have* managed to do it then this is the adventure for you. Priced at £5.50 it's value for money, and certainly a great deal of fun to play.

Computacalc ZX
Silicon Tricks
Darren - John Norbury

In the world of the commercially-used computer, where the large mainframe machines produced by such companies as IBM and ICL reign supreme, the price paid for a decent business program could buy a good family micro. Luckily, Sinclair programs come a little cheaper than this. In a seemingly games-dominated market it is good to see that the utility program has not been neglected.

Such a program is Computacalc ZX, produced by the software company Silicon Tricks. The purpose of this program, in general, is to provide a table from which data can be stored and manipulated. To use a simple example: in a commercial setting, sales and costs for a company could be entered into two separate boxes for each month and the computer will calculate the profit which it will display in a third box. The above is a very simple example. The possible uses for this program are really limitless.

Calculations are made using formulae entered by the user, but formulae in the simple form of, say, let the contents of box 3 = the sum of boxes 1 and 2 (which would actually be written as F0203 + 0303, assuming that box 1 is at grid reference 0203 and box 2 is at grid reference 0203).

The table which forms the display is actually much larger than the screen area itself so by a little dexterous wizardry on the part of the user, certain areas of the table are scanned.

Computacalc ZX gets full marks from me as a household, scientific and small business utility program. The only disadvantage I can find in the package lies in the seven pages of instructions. If you are a fairly new computer user then you may be a little overwhelmed at the complicated way in which the instructions are laid out. However, once deciphered — never forgotten!

Computacalc ZX is available in some branches of W H Smiths and computer software shops.

Readers' hardware reviews

Not content with reviewing software, two readers have branched out and reviewed some ZX hardware.

ZONX-81
Bi-Pak Semiconductors
MD Brunt

Last December I decided that it would be nice to have some sound for my ZX81. Of the three sound synthesisers that were on the market, the ZONX was the cheapest, just, and it was a self contained unit, the others required an amplifier and a speaker although they did have two I/O ports.

My decision made, I sent off and prepared for a Sinclair style wait. I need not have worried. Although it was less than a month till Christmas the ZONX arrived in less than one week! I quickly tried to fit the ZONX to my ZX81, only to discover that my full size keyboard got in the way. The problem was that there is a small volume control on the front of the ZONX and it was this that prevented the ZONX from fitting. My solution was to fit a new control into the top of the ZONX.

I opened the ZONX to find a speaker and several chips, all soldered very neatly I might add, and the AY-3-8912. The volume control turned out to be a preset pot, which is not really designed for a lot of use, and I would suggest that it should be moved as little as possible to reduce wear. I replaced it with a standard 'volume control' which I mounted through the top of the ZONX.

The manual for the ZONX is small (5½" by 4") and only twenty pages long, but it is fairly clear and it explains what each of the AY-3-8912's thirteen registers are for and how they are used. The ZONX resides in the I/O area of the Z-80's memory and so a short (9 bytes) machine code routine is needed to access it. The routine is held in a REM statement and can be typed in directly from the keyboard. The manual also contains eight short programs in- ▶

cluding one which turns the top row of keys (1-0) into an electronic organ.

The 16K RAM pack connects to the ZX81 through the ZONX, the edge connector of the ZX81 being duplicated on the ZONX, which is just as well since a simple explosion takes up twenty-five lines of BASIC.

The main disadvantage of the ZONX, or any sound synthesiser, is the time needed to develop a given sound – in most cases done by trial and error. Since neither Bi-Pak nor any other software company that I know of, markets any software for the ZONX, you will have to work on them yourself. The results can be worth it though. The other problem is with speed, most sounds are built up from BASIC loops and so they must either be run in fast mode or they sound far too slow. To use sound in games, therefore, will require some simple machine code programming.

I have had a lot of fun with my ZONX and I would recommend it to anyone, provided that they are prepared to spend a lot of time with it. If not, then they would be wasting their money. The ZONX units cost £25.95 for the ZX81 and Spectrum. (An extension board is required for the Spectrum at £6.80).

Memotext
Memotech
J Lowther

Clive Sinclair's 12 oz piece of modern technology that all the 'big boy' computers used to laugh at has been made to do some pretty amazing things by clever software and additional hardware, but word processing?

The functions in Memotext include keyboard input at normal speed, justification to left only (as in a letter) or left and right margins (as in *ZX Computing*) search/exchange/move groups or single words, lower case, headings, page numbers, saving files, pre-entered dates, prompts before printing and so on.

For instance, if you run a newsagents, you might set up data files of customers' names and what they owe you and then get the computer to print a set letter saying, basically, "can I have my money please?" You would insert the relevant names, addresses and amounts due each time. You could then use another text file (ie set letter) and use prompts to enable you to print orders for the wholesalers according to which special offers you've spotted in the "Newsagents Weekly"!

The Memotext is well documented although it took me about four days to get the hang of all the functions. Now I can type as badly as I like, check for spelling mistakes (on the screen) and correct or change any words that don't seem right at the press of a button, move around sentences so that they follow a logical sequence and then print out as many 'personalised' copies as I like, marked for the relevant people.

To be of use, you would really need a proper printer interface (Centronics or RS232) and access to a printer. This means that the 16K '81, Word Processor, and interface together would probably set you back about £130. For comparison, a 16K BBC with interface and wp would cost perhaps £330 and a commercial wp package probably little short of £1000.

The Memotext, like all the Memotech range, comes in a firm aluminium case that plugs directly into the ZX81 without wobble and fits between your RAM pack and '81. This means that you don't waste time loading the program.

Obviously, there are limitations: the '81 can't display the 60-80 characters a line commonly used in letters and reports so the Memotext uses a clever "word-wrap" feature which means you never lose words at the line ends but you have to visualise the final document rather than see it straight in front of you. Also some of the symbols and lower case letters make entering the text a bit complicated.

The Memotext is £29.90 (including VAT and p&p). This compares with about £8 for some cassette-based systems (eg Tasword or Contrast Software's Text, £6). The buyer must judge (a) whether they really need word processing on their '81, (b) whether they would use it often (printers will often do wp at about £2 a sheet – if you only want say one a month, it might be cheaper) and (c) whether they need the extra speed, convenience and general 'slickness' of a ROM-based system.

Club corner



South African Computer Club

Dear ZX Computing
I would be pleased if you would publish this letter in your magazine. I am the European representative for a South African based computer club (ZX Computers). We are a world wide club and have members in Australia, Europe and, of course, South Africa.

Although we are a big club in South Africa, and have members in Britain and Portugal, we would like some more members in Europe where the response so far has been relatively poor. We would also like some American members. Anyone interested should write to the address below and I shall send them further information.
Yours faithfully

J. Jones
European Representative
House 14
Anglo Alpha
PO Box 15
1725 Roodepoort
South Africa

Australian ZX Users Association

Dear ZX Computing
The Australian ZX Users Association has been in existence since late 1981 when we first supported the ZX80. Since then we have expanded to be one of the largest ZX groups in Australia and now support all the Sinclair machines. We produce a bi-monthly newsletter of some twenty pages and organise local users meetings around Australia. By rights we should be called the International ZX Users Club as we have many members from around the world.

A subscription to our newsletter and the other facilities we provide costs A$9 pa for Australian members and A$15 for international membership.

For further information write to the address below enclosing A40 c stamp or three International Reply Coupons.
Yours faithfully

David Vernon
Secretry AZUA
19 Godfrey Street
Campbell ACT
2601
Australia

Anglo French Computing?

Dear ZX Computing
Can you help me? I'm a French 48K Spectrum owner who would like to exchange ideas and programs with British users. Are there any clubs or groups that correspond with European users?

If so, please write to the address below as my Spectrum is thirsty for programs.
Yours faithfully

Patrick Dardour
18 Rue du Borrego
75020 Paris
France

Turkish ZX Club?

Dear ZX Computing
I'm Turkish and live in Istanbul and I own a 48K Spectrum and a ZX Printer. I would like to become a member of a ZX club but unfortunately there isn't one around here that I know of.

I would like, therefore, to start a new club: I have a few supporters (aged 17 to 18) but we would also like to have pen pals who would be interested in becoming members of such a club. We would appreciate it if you would publish this letter and help us with our idea.
Yours faithfully

Edip Uras
Mühürdar cad No 91/4
Kadikoy
Istanbul
Turkey

The Midweekly Spectrum User Group

Dear ZX Computing
I would like to tell your readers about the Midweekly Spectrum User Group. We meet every Wednesday (or Thursday if Wednesday's date is odd) at 8 Linnet Close, Gloucester anytime after 7 pm. Naturally, as there is no charge, members are invited at their own risk.

Members may bring their own Spectrum and cassette recorder to the meeting and use of the televisions is on first-come-first-served basis, as is the light pen and tape to tape facility.

The aim is to have a congenial evening, with a mutually rewarding exchange of ideas and knowledge, so age, experience or gender is unimportant. For more details, telephone me on Gloucester 23186.
Yours faithfully

Barry Ledbury

ZX Club Bangalore

Dear ZX Computing
I am writing to inform you of our ZX Club in Bangalore. This is a newly formed club with around 16 members owning Sinclair computers. We meet every week on Sundays between 3 and 5pm to exchange news, software and ideas. I am subscribing *ZX Computing* for the club as it is very informative for ZX users.

I would very much like to exchange news and ideas with other ZX users and groups and would be grateful for any information that could be sent to us at the address below.
Yours faithfully

B S Srinivasan
No36 First Main Road
N R Colony
Bangalore 560019

Acocks Green Computer Club

Dear ZX Computing
The Acocks Green Computer Club meets every third Thursday in the month at St Mary's Church (kitchen vestry) at 7 pm.

I also run a computer users group that meets every Monday at 7pm at the Stonehall Adult Education Centre during term time. Because the meetings are held in an educational establishment, no restriction is put on the machine but obviously Spectrums abound.

Please contact me at the address below or at either venue for further informaton.
Yours faithfully

Mike Bedford-White
16 Westfield Road
Acocks Green
Birmingham
B27 7TL

If you run, or are a member of a user club which caters for the Sinclair user, why not get your group on the map by writing to us at:

Club Corner,
ZX Computing,
1 Golden Square,
London W1R 3AB

All you have to do is to send us a letter with details of your club's activities (times of meetings, addresses of who to contact etc.) and we'll do the rest. If you publish a newsletter or club magazine, we'd very much like to see that too.

And if you don't see a club in your area, why not start one up by writing to *ZX Computing* and seeing if any like-minded enthusiasts wish to join you.

Racer

Move into top gear and try your hand at Grand Prix racing courtesy of David Price of Cardiff.

"Well, here we are at the start of the Spectrumapolis 500 where of course the hot favourite is . . . ." (Insert your own name here.) Type in this game and you'll soon be driving round and round the course and through the chicanes trying to finish in the best possible time. Crashing into the barriers tends to be slightly fatal.

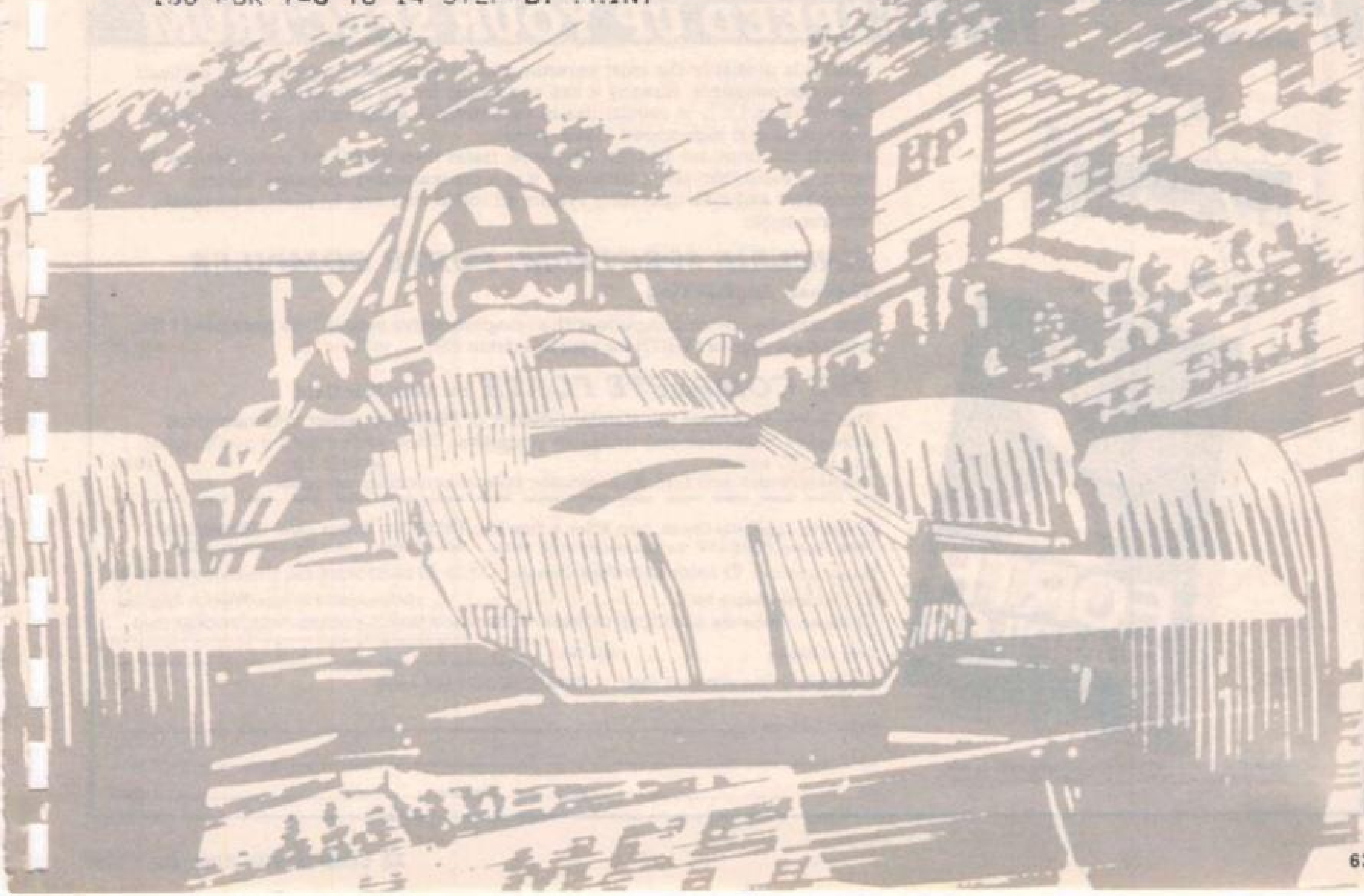
The sound and graphics capabilities of the Spectrum are put to good use in this game and, as there is a "best time" feature, it becomes quite addictive as well as fun to play.

Well then, get typing, put on your crash helmet and push your foot to the floor. Instructions and controls are included in the program. NB. To qualify for the championship you will have to beat David's best score of 32 seconds, yours truly only managed 37!

```
   1 GO SUB 1000: LET a=2: BORDE
R 1: PAPER 1: CLS
   3 PRINT  AT 8,6; INK a;"███ █
██ ███ ███ ███                 █ █ █
 █ █   █   █ █                 ███ █
██ █   ███ ███                 ██  █
 █ █   █   ██                  █ █ █
 █ ███ ███ █ █"
   5 IF a=8 THEN  BEEP .3,12: BE
EP .5,6: BEEP .3,15: GO TO 10
   6 LET a=a+1: FOR f=1 TO 4: LE
T o=.0123: BEEP o,2: BEEP o,4: B
EEP o,6: BEEP o,8: BEEP o,10: BE
EP o,12: BEEP o,14: BEEP o,16: B
EEP o,2: BEEP o,4: BEEP o,6: NEX
T f: GO TO 3
  10 INK 7: CLS : LET s=0: LET t
ime=0
  15 PRINT  AT 0,12;"RACER"'''"(A
)-- LEFT","(S)-- RIGHT"''"(K)--
UP","(M)-- DOWN"
  17 PRINT '''THIS IS A RACE AGA
INST THE CLOCK"
  20 PRINT '''       @ DAVID PRIC
E 1983"''''"PRESS ANY KEY TO BEGI
N THE RACE ": PAUSE 0
  51 REM ***********************
  52 REM * The "A" in lines     *
  53 REM * 210,240 and 260 are *
  54 REM * entered in GRAPHICS *
  55 REM * mode.                *
  56 REM ***********************
  90 LET q=1: LET la=100: LET la
p=1: BORDER 1: PAPER 1: INK 9: C
LS
 100 FOR f=0 TO 31: PRINT  AT 1,
f; PAPER 2;"*"; AT 20,f;"*";: NE
XT f
 101 FOR f=1 TO 20: PRINT  AT f,
0; PAPER 2;"*"; AT f,31;"*";: NE
XT f
 103 FOR f=3 TO 19: PRINT  AT f,
3; PAPER 6;"*"; AT f,13;"*"; AT
f,19;"*"; AT f,28;"*": NEXT f
 104 FOR f=2 TO 18: PRINT  AT f,
6; PAPER 6;"*"; AT f,16;"*"; AT
f,25;"*": NEXT f
 105 FOR f=7 TO 10: PRINT  AT 6,
f; PAPER 3;"*"; AT 12,f;"*"; AT
18,f;"*": NEXT f
```

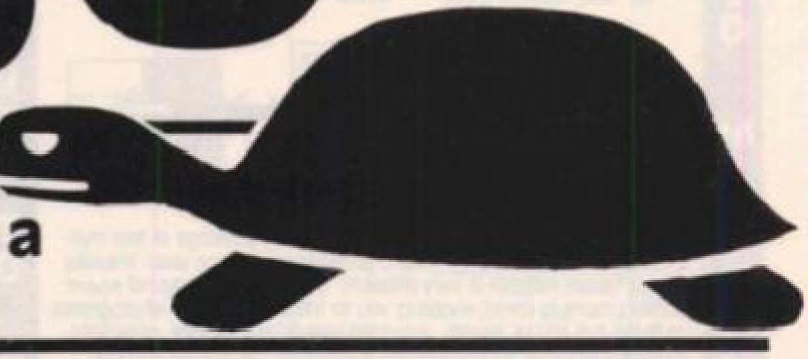


```
 106 FOR f=9 TO 12: PRINT  AT 3,
f: PAPER 3;"*"; AT 9,f;"*"; AT 1
5,f;"*": NEXT f
 107 FOR f=20 TO 22: PRINT  AT 3
,f; PAPER 4;"*"; AT 9,f;"*"; AT
15,f;"*": NEXT f
 108 FOR f=22 TO 24: PRINT  AT 6
,f; PAPER 4;"*"; AT 12,f;"*"; AT
 18,f;"*": NEXT f: GO TO 190
 120 FOR f=3 TO 15 STEP 6: PRINT
  AT f,8; PAPER 3;"*"; AT f,23;
PAPER 4;"*";: NEXT f
 121 FOR f=6 TO 18 STEP 6: PRINT
  AT f,11; PAPER 3;"*"; AT f,21;
 PAPER 4;"*";: NEXT f: GO TO 200
 140 FOR f=4 TO 16 STEP 6: PRINT
  AT f,8; PAPER 3;"*"; AT f,23;
PAPER 4;"*";: NEXT f
 141 FOR f=5 TO 17 STEP 6: PRINT
  AT f,11; PAPER 3;"*"; AT f,21;
 PAPER 4;"*";: NEXT f: GO TO 200
 160 FOR f=3 TO 18 STEP 6: PRINT
  AT f,2; PAPER 6;"*"; AT f,5;"*
"; AT f,14;"*"; AT f,17;"*"; AT
f,26;"*"; AT f,29;"*";: NEXT f
 161 FOR f=6 TO 18 STEP 6: PRINT
  AT f,1; PAPER 6;"*"; AT f,4;"*
"; AT f,15;"*"; AT f,18;"*"; AT
f,27;"*"; AT f,30;"*";: NEXT f:
GO TO 200
 180 FOR f=8 TO 14 STEP 6: PRINT
  AT f,8; PAPER 3;"*"; AT f,23;
PAPER 4;"*";: NEXT f
 181 FOR f=7 TO 13 STEP 6: PRINT
  AT f,11; PAPER 3;"*"; AT f,21;
 PAPER 4;"*";: NEXT f: GO TO 200
 194 POKE 23673,255: POKE 23672,
255
 196 PRINT  AT 20,29; INK 1;"+";
 AT 20,30;"+"
 198 PRINT  AT 0,0;"FASTEST TIME
 FOR COURSE "; INT (time); AT 21
,0;"TIME"; AT 21,16;"LAP"
 200 PRINT  AT 21,20;lap: LET a=
19: LET b=1
 205 LET ti=(256* PEEK 23673+ PE
EK 23672)/50
 208 PRINT  AT 21,6; INT ti
 210 PRINT  AT a,b;"A"; AT a,b;"
A"; AT a,b;"A"; AT a,b;" ";
 220 LET b=b-( INKEY$ ="a")+( IN
KEY$ ="s"): BEEP .001,3
 230 LET a=a-( INKEY$ ="k")+( IN
KEY$ ="m")
 240 IF  SCREEN$ (a,b)="*" THEN
 FOR q=0 TO 7: PRINT  AT a,b; FL
ASH 1;"A": BORDER q: BEEP .09, R
ND *20: NEXT q: GO TO 700
 250 IF  SCREEN$ (a,b)="+" THEN
```



```
BEEP .5,6: GO TO 500
260 PRINT  OVER 1; AT a,b;"A"
300 GO TO 205
500 LET la=la+20: LET lap=lap+1
502 IF lap=6 THEN  GO TO 600
504 PRINT  AT 21,20;lap
506 GO TO la
600 FOR f=0 TO 7: BEEP .002, RND : BEEP .003, RND : BEEP .008, RND : BEEP .006, RND : BORDER f: NEXT f: BORDER 7: PAPER 7: INK 0: CLS
610 PRINT "YOU HAVE COMPLETED THE RACE"''''"YOU FINISHED IN "; INT (ti);" SECONDS"
611 LET s=s+1
612 IF s>1 THEN  GO TO 630
615 IF  INT (ti)>time THEN  LET time= INT (ti)
620 PRINT '''"BEST TIME FOR COURSE "; INT (time)
625 GO TO 730
630 IF  INT (ti)<time THEN  LET time= INT (ti)
640 GO TO 620
700 BORDER 7: PAPER 7: INK 0: CLS
710 PRINT "YOU CRASHED INTO THE BARRIERS"'''"YOU RACED FOR "; INT (ti);" SECONDS"
720 PRINT '''"YOU WERE ON LAP ";lap'''"BEST TIME FOR COURSE ";time
730 PRINT ''''"    ANOTHER RACE (Y::N)"
731 IF  INKEY$ ="Y" OR  INKEY$ ="y" THEN  GO TO 50
732 IF  INKEY$ ="N" OR  INKEY$ ="n" THEN  GO TO 2900
733 GO TO 731
1000 FOR a=0 TO 7: READ n: POKE  USR "A"+a,n: NEXT a: RETURN
1010 DATA 60,60,60,255,255,255,102,102
2900 BORDER 7: PAPER 7: CLS : LET q=1
3000 PRINT  AT 10,12; INK q;"CHERIO"
3003 IF q=6 THEN  LET q=1
3004 LET q=q+1: FOR f=1 TO 4: LET a=.012: BEEP a,-4: BEEP a,-6: BEEP a,-8: BEEP a,-10: BEEP a,-12: BEEP a,-14: BEEP a,-16: BEEP a,-2: BEEP a,-4: BEEP a,-6: NEXT f: GO TO 3000
5000 SAVE "RACER" LINE 1
```

Slogo



David Nowotnik presents the first of a three-part series on a fascinating language.

In the last issue of *ZX Computing*, Tim Hartnell presented an excellent article introducing the programming language of LOGO. I'm sure many readers will now want to try LOGO on their Spectrums. But, if you've tried to find some commercial software which allows you to use LOGO then you might well have been disappointed; there isn't much available. As a remedy to that situation, I shall be presenting, in a series of three articles, a BASIC program which simulates LOGO. Don't worry if you missed Tim's article, as I'll be going over the principles of LOGO again.

In each part of the series, you'll get a BASIC program to type into your Spectrum, and instructions on how to use the program. And, in using the program, you'll be using LOGO. So, by the time the series is complete, you should have a good understanding of the language. The program I'll be presenting next time must be added or MERGEd to the program in this issue, then the one in Part 3 added to the other two. So, while this month's program will fit a 16K Spectrum, you'll need a 48K Spectrum to accommodate the final version.

Ideally, a LOGO translation program should be written in machine code for a fast operating speed. I've written my version in BASIC as it's easier for me to write it, and easier for you to type in. But in using BASIC, speed is lost — that's why I've called my version of this language SLOGO!

Once you have entered the listing in Fig.1 and SAVEd the program on tape, you are ready to start. But before we begin typing in our LOGO commands, it might be useful to consider the background and principles of this language.

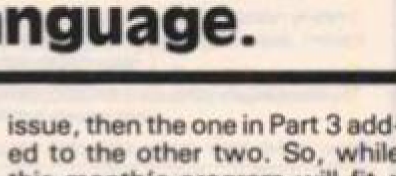


Fig. 1. The program listing

```
   5 REM ************************
  10 REM *                      *
  15 REM *        SLOGO         *
  20 REM *                      *
  25 REM *  by David Nowotnik   *
  30 REM *    January, 1984     *
  35 REM *                      *
  40 REM ************************
  45 REM
  50 REM      Initialise
  55 REM
  60 READ m: READ n: READ o
  65 DIM x$(m,2): DIM w$(n,12):
 DIM f$(o,12)
  70 DIM u(m): DIM v(n): DIM g(o
 )
  75 LET pp=1: LET wr=1: LET tur
t=1
  80 FOR i=1 TO m: READ x$(i)
  85 READ u(i): NEXT i
  90 FOR i=1 TO n: READ w$(i)
  95 READ v(i): NEXT i
 100 FOR i=1 TO o: READ f$(i)
 105 READ g(i): NEXT i
 110 DATA 6,8,1
 115 DATA "FD",3000,"BK",3200,"L
T",3400,"RT",3600
 120 DATA "PU",4000,"PD",4050
 200 DATA "FORWARD",3000,"BACK",
3200,"LEFT",3400
 205 DATA "RIGHT",3600,"DRAW",38
00,"HOME",3850
 210 DATA "PENUP",4000,"PENDOWN"
,4050
 300 DATA "RANDOM",8000
 400 POKE 23658,8
 480 GO SUB 3800: GO TO 2000
 490 REM
 500 REM   Error subroutines
 510 REM
 520 GO SUB (690+err*10)
 530 PRINT #1;a$
 540 PAUSE 250
 550 RETURN
 700 LET a$="Command error - re-
enter the     line.": RETURN
 710 LET a$="Number error - re-e
nter the line": RETURN
 720 LET a$="No wrap - the line
cannot be     drawn": RETURN
 990 REM
1000 REM   Subroutines
1010 REM
1020 REM   Entry check
1030 LET t=0: LET y$=""
1040 LET s=s+1: IF s>LEN z$ THEN
 LET t1=1: RETURN
1050 IF (z$(s)=" " OR z$(s)=CHR$
```

```
 Ø) AND t=Ø THEN  GO TO 1Ø4Ø
1Ø6Ø IF (z$(s)=" " OR z$(s)=CHR$
 Ø) AND t=1 THEN  RETURN
1Ø7Ø LET y$=y$+z$(s): LET t=1: G
O TO 1Ø4Ø
1Ø8Ø REM
11ØØ REM  Draw/Erase turtle
111Ø IF x<3 OR x>252 OR y<3 OR y
>172 THEN  RETURN
112Ø FOR j=16Ø TO 2ØØ STEP 4Ø
113Ø LET q=dir+j: IF q<Ø THEN  L
ET q=36Ø-j
114Ø IF q>36Ø THEN  LET q=q-36Ø
115Ø LET q=q*PI/18Ø
116Ø LET x1=5*SIN q: LET y1=5*CO
S q
117Ø PLOT  INVERSE turt;x,y
118Ø DRAW  INVERSE turt;x1,y1
119Ø NEXT j: RETURN
12ØØ REM
121Ø REM  A number or a function
122Ø GO SUB 1Ø2Ø: IF err=1 THEN
 LET err=2: RETURN
123Ø IF CODE y$>57 THEN  GO TO 1
3ØØ
124Ø IF y$="" THEN  LET err=2: R
ETURN
125Ø FOR k=1 TO LEN y$
126Ø IF CODE y$(k)<48 OR CODE y$
(k)>57 THEN  LET err=2
127Ø NEXT k
128Ø IF err=Ø THEN  LET a=VAL y$
129Ø LET t=Ø: RETURN
13ØØ REM  Evaluate a function
131Ø LET y$=(y$+"          ")(
 TO 12)
132Ø FOR i=1 TO o: IF y$=f$(i) T
HEN  GO TO 134Ø
133Ø NEXT i: LET err=2: RETURN
134Ø GO SUB g(i): RETURN
14ØØ REM  X Wrap
141Ø LET cr=1: IF q<2*PI AND q>P
I THEN  LET cr=-1
142Ø IF cr=1 THEN  LET x1=255-x:
 IF q>1.57 THEN  LET y1=Ø: LET x
2=Ø: GO TO 143Ø
1425 IF cr=1 THEN  LET y1=x1/TAN
 q: LET x2=Ø
143Ø IF cr=-1 THEN  LET x1=-x: L
ET y1=(x1/TAN (q-PI)): LET x2=25
5
144Ø LET y2=y+y1: LET a=a-INT SQ
R (ABS (x1)^2+ABS (y1)^2)
145Ø RETURN
147Ø REM
15ØØ REM Y Wrap
151Ø LET cr=1: IF q>PI/2 AND q<3
*PI/2 THEN  LET cr=-1
152Ø IF cr=1 THEN  LET y1=175-y:
 LET x1=y1*TAN q: LET y2=Ø
153Ø IF cr=-1 THEN  LET y1=y: LE
T x1=-(y1*TAN (q-PI)): LET y2=17
5: LET y1=-y
154Ø LET x2=x+x1: LET a=a-INT SQ
R (ABS (x1)^2+ABS (y1)^2)
155Ø RETURN
16ØØ REM X and/or Y WRAP
16Ø5 IF q>PI/2 THEN  GO TO 163Ø
161Ø LET x3=x+(175-y)*TAN q
1615 IF x3>255 THEN  GO TO 14ØØ
162Ø IF x3<255 THEN  GO TO 15ØØ
1625 GO TO 168Ø
163Ø IF q>PI THEN  GO TO 1645
1635 LET x3=x+y*TAN (PI-q)
164Ø GO TO 1615
1645 IF q>3*PI/2 THEN  GO TO 167
Ø
165Ø LET x3=x-y*TAN (q-PI)
1655 IF x3<Ø THEN  GO TO 14ØØ
166Ø IF x3>Ø THEN  GO TO 15ØØ
1665 GO TO 168Ø
167Ø LET x3=x-(175-y)*TAN (2*PI-
q)
1675 GO TO 1655
168Ø IF y1>=x1 THEN  LET a1=INT
(y1/COS q+1.5)
1685 IF y1<x1 THEN  LET a1=INT (
x1/SIN q+1.5)
169Ø LET a=a-a1: GO SUB 145Ø: IF
 y2<Ø THEN  LET y2=175
1695 IF y2>175 THEN  LET y2=Ø
17ØØ GO SUB 15ØØ: IF x2<Ø THEN
LET x2=255
17Ø5 IF x2>255 THEN  LET x2=Ø
171Ø RETURN
199Ø REM
2ØØØ REM  Input routine
2Ø1Ø REM
2Ø2Ø INPUT "W:"; LINE z$: LET s=
Ø
2Ø25 LET count=Ø: LET rc=Ø: GO S
UB 2Ø4Ø: GO TO 2Ø2Ø
2Ø3Ø REM
2Ø4Ø REM  Command check
2Ø5Ø REM
2Ø6Ø LET t=Ø: LET t1=Ø: LET err=
Ø
2Ø7Ø GO SUB 1Ø2Ø
2Ø8Ø IF t1=1 AND y$="" AND count
=Ø THEN  LET turt=Ø: GO SUB 11ØØ
2Ø85 IF t1=1 AND y$="" THEN  RET
URN
2Ø9Ø IF LEN y$<>2 THEN  GO TO 21
3Ø
21ØØ FOR i=1 TO m
211Ø IF y$=x$(i) THEN  GO TO 22Ø
Ø
212Ø NEXT i
```

```
2130 LET y$=(y$+"          ")(
 TO 12)
2140 FOR i=1 TO n
2150 IF y$=w$(i) THEN  GO TO 221
Ø
2160 NEXT i
2170 REM  Command error
2180 LET err=1: IF turt=1 THEN
LET turt=Ø: GO SUB 1100
2190 GO TO 520
2200 GO SUB u(i): GO TO 2220
2210 GO SUB v(i)
2220 IF err>Ø THEN  GO TO 520
2230 GO TO 2070
2990 REM
3000 REM   FORWARD routine
3010 REM
3020 GO SUB 1200: IF err>Ø THEN
 RETURN
3030 IF turt=Ø THEN  LET turt=1:
 GO SUB 1100
3040 LET q=dir*PI/180
3050 LET x1=INT (.5+a*SIN q): LE
T y1=INT (.5+a*COS q)
3060 LET tr=Ø: LET x2=x+x1: LET
y2=y+y1
3070 IF x2<Ø OR x2>255 THEN  LET
 tr=1
3080 IF y2<Ø OR y2>175 THEN  LET
 tr=tr+2
3090 IF tr>Ø AND wr=Ø THEN  LET
err=3: RETURN
3100 IF tr=Ø THEN  GO TO 3120
3110 GO SUB (1300+tr*100)
3120 IF pp=1 THEN  PLOT x,y: DRA
W x1,y1
3130 IF pp=Ø THEN  PLOT  INVERSE
 1; OVER 1;x,y: DRAW  INVERSE 1;
 OVER 1;x1,y1
3140 LET x=x2: LET y=y2: IF tr>Ø
 THEN  GO TO 3050
3150 LET t=Ø: RETURN
3190 REM
3200 REM  BACK
3210 REM
3220 IF turt=Ø THEN  LET turt=1:
 GO SUB 1100
3230 LET dir=dir+180: IF dir>360
 THEN  LET dir=dir-360
3240 GO TO 3020
3390 REM
3400 REM  LEFT
3410 REM
3420 IF turt=Ø THEN  LET turt=1:
 GO SUB 1100
3430 GO SUB 1200: IF err>Ø THEN
 RETURN
3440 LET dir=dir-a
3450 IF dir<Ø THEN  LET dir=360+
dir: GO TO 3450
3460 RETURN
3590 REM
3600 REM  RIGHT
3610 REM
3620 IF turt=Ø THEN  LET turt=1:
 GO SUB 1100
3630 GO SUB 1200: IF err>Ø THEN
 RETURN
3640 LET dir=dir+a
3650 IF dir>360 THEN  LET dir=di
r-360: GO TO 3650
3660 RETURN
3790 REM
3800 REM  DRAW
3810 REM
3820 CLS
3840 REM
3850 REM  HOME
3860 REM
3870 IF turt=Ø THEN  LET turt=1:
 GO SUB 1100
3880 LET x=128: LET y=88
3890 LET dir=Ø: LET turt=Ø
3900 GO SUB 1100: RETURN
3990 REM
4000 REM  PENUP
4010 REM
4020 LET pp=Ø: RETURN
4030 REM
4050 REM  PENDOWN
4060 REM
4070 LET pp=1: RETURN
4080 REM
8000 REM  RANDOM
8010 GO SUB 1210: IF err=2 THEN
 RETURN
8020 LET a=INT (RND*a): RETURN
```

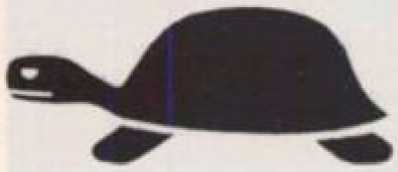
LOGO was designed to provide children with an early introduction to computer programming and to develop their abilities in logical thinking. The commands of LOGO are simple to understand, yet LOGO encourages a good programming style by virtue of its structure. LOGO is best known as a graphics language, enabling still (and animated) pictures and patterns to be drawn. More powerful versions of LOGO have facilities for text handling as well as graphics but my version, in common with most, deals with graphics only.

You'll find LOGO a very useful language if you have young children, if you want a versatile graphics routine or if you want to move on from BASIC, LOGO is an easy language to learn, yet it introduces you to structured programming, as used by more powerful languages such as FORTH.

Drawing with LOGO makes use of a turtle — but before you rush off with complaints to the RSPCA, these are only imaginary animals! Imagine you had precise control over the movements of your turtle; instructing it to go forward or back, turn left or right, all by specified amounts. Your turtle carries a pen which you can instruct to be lowered onto a sheet of paper, so that, as the turtle moves, a line is drawn. Using combinations of the four movement commands (forward, back, left, and right) drawings and patterns can be created.

Some versions of LOGO ac-

tually make use of a robotic 'turtle' which is interfaced to a computer. LOGO commands are transmitted to the turtle, creating shapes on a sheet of paper placed on the floor. More often, the monitor or TV sceen forms the paper and an electronic turtle is drawn and is moved, on the screen. This is how my version works.



RUN the program and you'll see the turtle on the screen. It is represented by the "Λ"; the point is its head and this indicates the direction of the turtle. You'll see that the turtle's starting point is at the centre of the screen, heading directly up the screen.

At the bottom of the screen you'll see "W;" and a flashing cursor. This tells you that the computer is waiting for you to enter a command. At this stage, this LOGO program will accept only 8 commands as shown in Fig 2. Note that all commands are entered in capital letters (CAPS LOCK mode is automatically set by the program) and for some commands, a two letter abbreviation can be used instead of the full word. Before you use any command, I'll describe what each one will do.

COMMAND	TWO LETTER ABBREVIATION
DRAW	
HOME	
FORWARD	FD
BACK	BK
LEFT	LT
RIGHT	RT
PENUP	PU
PENDOWN	PD

Fig. 2. The LOGO commands featured in this issue

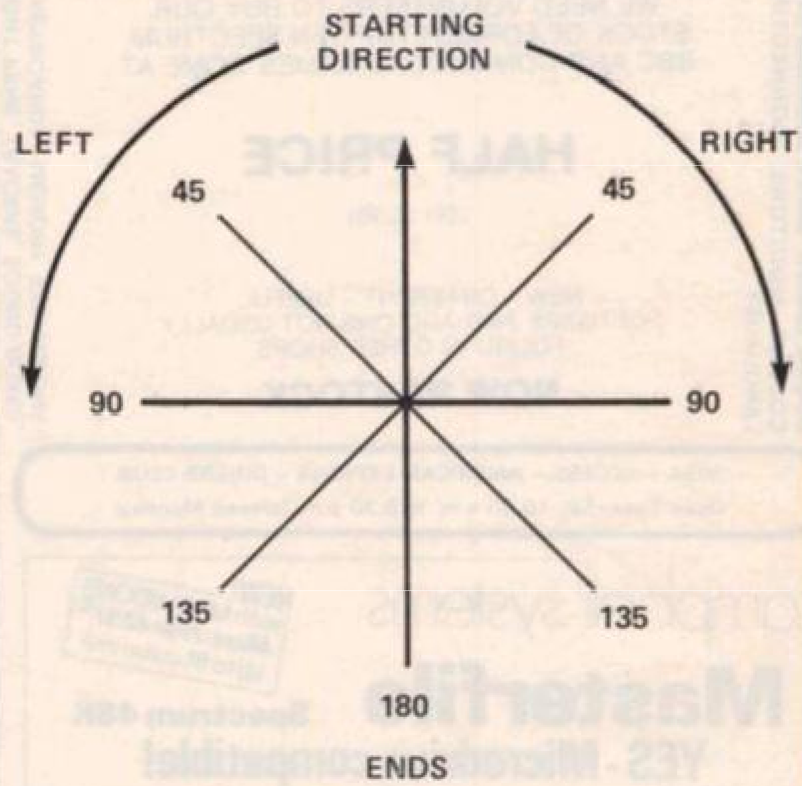


Fig. 3. A diagram of angles to turn the turtle

HOME — This moves the turtle from anywhere on the screen back to its starting position at the centre of the screen, heading directly up the screen.

DRAW — This clears the screen, then moves the turtle HOME.

FORWARD — The turtle is moved forward a specified amount. To complete the command, you have to tell the turtle how far to move forward, eg FORWARD 20 (or FD 20). The unit of distance is one pixel; remember, the screen is made up 256 pixels across, and 176 down.

BACK — The turtle is turned head-to-tail, then moved in the same way as FORWARD (eg BACK 25).

LEFT — The turtle's head remains in the same place but its body will swivel so that its direction rotates to the left. You have to tell the turtle how far to turn. As do FORWARD and BACK, the command LEFT requires a number added to it to complete the instruction, to tell the turtle how much to turn. This number is the angle of turn in degrees. If you've forgotten that 90 degrees makes a right angle, then use Fig 3 as a guide when using the LEFT (or RIGHT) command. Angles should be whole numbers between 0 and 360.

RIGHT — This is the same principle as LEFT, but, of course, the turtle turns right instead of left.

PENUP — This command raises the pen from the paper so that the turtle can be moved without a line being drawn.

PENDOWN — The pen is placed on the paper.

Now, lets try a few examples. Type in:

FORWARD 40 (or FD 40), and press enter. You'll see the turtle disappear, a line 40 pixels long will be drawn, and then the turtle reappears. To enable the program to work as quickly as possible, our turtle will always disappear when in motion and will only reappear when all instructions are complete and the "W:" symbol comes back on the screen. Also, notice that when the program starts, the pen is in the down position, enabling a line to be drawn.

Now, lets turn the turtle 90° to the right. Type in:

RIGHT 90 (or RT 90) and press enter.

Then to move forward again:

FD 40.

If you continued entering alternatively RT 90 and FD 40, you would end up with a square. As it is cumbersome to type in one command at a time, LOGO allows you to string commands together. Let's see how. First, clear the screen, and reset the turtle using the command DRAW, then enter:

FD 40 RT 90 FD 40 RT 90 FD 40 RT 90 FD 40 RT 90 and enter.

Leave a single space between each command and number. This, again, draws a square. You may have noticed that we are repeating the same two commands four times. Fortunately, LOGO allows repetition of commands to be simplified and it's one of the things I'll be dealing with next time.

What should happen if the turtle goes off the screen? Let's find out. Clear the screen with DRAW, then enter:

RT 10 FD 2000.

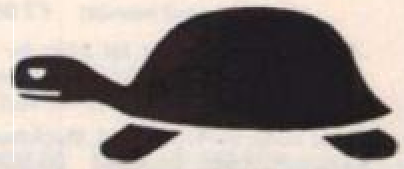
The turtle is turned slightly to right, then moves forward 2000 pixels. As it disappears off the edge of the screen, our turtle reappears on the other side. This feature is called wrap. It is useful to draw special patterns and means that you won't get an error message if you accidentally go off the edge of the screen.

Another feature of this LOGO program is that the specified number used by FORWARD, BACK, RIGHT and LEFT can be replaced by a random number. As an example, try:

FD RANDOM 50

The turtle will move forward by an amount somewhere between 0 and 50.

That's enough of the theory of LOGO for this time. Try making up some drawings, patterns, or shapes for yourself. The program has an error trapping routine, so if you make a mistake, you should get a reasonably friendly error message. In the next issue of ZXC, you'll notice a quantum leap in the power of this LOGO translater.



BUFFER MICRO SHOP

310 STREATHAM HIGH ROAD, LONDON SW16 6HG
Tel: 01-769 2887

BOOKS, GAMES, SMALL BUSINESS SYSTEMS, EDUCATIONAL, HOUSEHOLD, COLOUR MONITORS, CONNECTING WIRES, COMPUTING AIDS, LANGUAGES.

HARDWARE AND ACCESSORIES, LIGHT PENS, TRACERS, SOUND BOXES, PRINTERS, V.D.U.'s, INTERFACES, KEYBOARDS, MEMORY EXPANSIONS.

THE ORIGINAL "SINCLAIR" SPECIALISTS

HELP

WE NEED VOLUNTEERS TO BUY OUR STOCK OF FORMER TOP-TEN SPECTRUM, BBC AND COMMODORE GAMES, SOME AT

HALF PRICE

(OR LESS)

NEW – DIFFERENT – USEFUL SOFTWARE AND ADD-ONS NOT USUALLY FOUND IN OTHER SHOPS

NOW IN STOCK

VISA – ACCESS – AMERICAN EXPRESS – DINERS CLUB
Open Tues.-Sat. 10.30 a.m. to 5.30 p.m. (closed Monday)

REGARDEZ

LANGUAGE LEARNING AIDS FOR FRENCH, GERMAN & SPANISH

FOR BBC(32K) · ELECTRON SPECTRUM (48K)

As used in numerous schools and colleges these programs provide a highly successful aid to modern language learning. Each cassette contains a sophisticated control program and a comprehensive series of vocabulary lessons which can be used in a variety of self-paced learning and test modes. Words, phrases etc are displayed with all necessary accents and special characters, different colours are used for masculine, feminine and neuter words to assist gender learning.

The programs are suitable for beginners, 0-level and beyond as simple commands enable new lessons in vocabulary or grammar to be created by the user, edited as required, then permanently saved for later use. Invaluable for homework and exam revision!

Two cassettes are available for each language, together these contain a vocabulary of thousands of words; Level A provides 16 lessons in general subjects; Level B provides a further 16 lessons including adjectives, adverbs and fully conjugated verb lists.

Available from your computer store or by mail order Price £9.95
Also Available "ANSWER BACK General Knowledge Quiz" Price £10.95



KOSMOS SOFTWARE

KOSMOS Software 1 Pilgrims Close, Harlington, DUNSTABLE, Beds. LU5 6LX
Telephone (05255) 3942

Please supply the following programs

The French Mistress Level A = £9.95 ☐	The French Mistress Level B = £9.95 ☐
The German Master Level A = £9.95 ☐	The German Master Level B = £9.95 ☐
The Spanish Tutor Level A = £9.95 ☐	The Spanish Tutor Level B = £9.95 ☐
ANSWER BACK Quiz (Senior) = £10.95 ☐	

I have a BBC/Electron/Spectrum computer (delete as necessary)

Mr/Mrs/Miss

Address

Post code

I enclose a cheque/postal order for £ ______ payable to KOSMOS Software

KOSMOS SOFTWARE
1 Pilgrims Close, Harlington, DUNSTABLE, Beds. LU5 6LX

campbell systems

NEW! with MYRMIDON'S Micro-Print 42/51© up to 51 columns!

Masterfile — Spectrum 48K

YES - Microdrive compatible!

Acclaimed as the definitive filing system for the 48K Spectrum – MASTERFILE'S machine coded flexibility gives you 32K (max) per file – 26 fields per record – up to 128 characters per field – multiple level searches for numeric or character comparisons – data presentation in any one of 36 user-defined displays which may be sequenced by any field – USER-BASIC for tailored processing. **" . . . the most comprehensive of the data-bases . . . "** Sinclair User June 1983

With example file and detailed manual £15.00

Drawmaster — Spectrum 48K

DRAWMASTER is the ultimate DRAW AND PAINT utility for the Spectrum – compose your own HI-RES pictures with 8 direction 2 speed hand draw keys – fast fill – enlarge/reduce facility for all or part of the screen plus all the Spectrum colours and attributes – over 50 commands in all.

With detailed instructions £6.95

DLAN DISPLAY LANGUAGE — Spectrum 16/48K

DLAN ('dee-lan') DISPLAY LANGUAGE is a powerful machine coded interpreter for generating DYNAMIC TEXTUAL DISPLAYS for advertising and education – all commands are single letter or symbol – with 4-way scroll of 11 type-faces through user-defined windows.

With detailed manual £7.95

All programs mailed 1st class by return. Prices include VAT and postage within Europe. SAE for full list.

Dept.(ZX) 15 Rous Road, Buckhurst Hill, Essex, IG9 6BL England 01 504 0589

Zappo

Use this program by E French of Lancaster to get super sounds for your games.

Anyone who has tried to write their own games on the Spectrum will have found the BEEP command very limiting and many people have bought add-ons to raise the Spectrum's sound to that of the BBC micro in versatility as well as volume. However, for most people £20-£30 is a high price for such a simple task.

Fortunately, there is a compromise solution which uses software rather than hardware to achieve special effect sounds such as explosions at negligible cost.

There is, of course, the inevitable snag. Unlike the better add-ons, execution of the program stops while the sound is being generated. However, the program does contain three pre-defined sounds:

1) A Zap
2) A Machine gun shot
3) An Explosion.

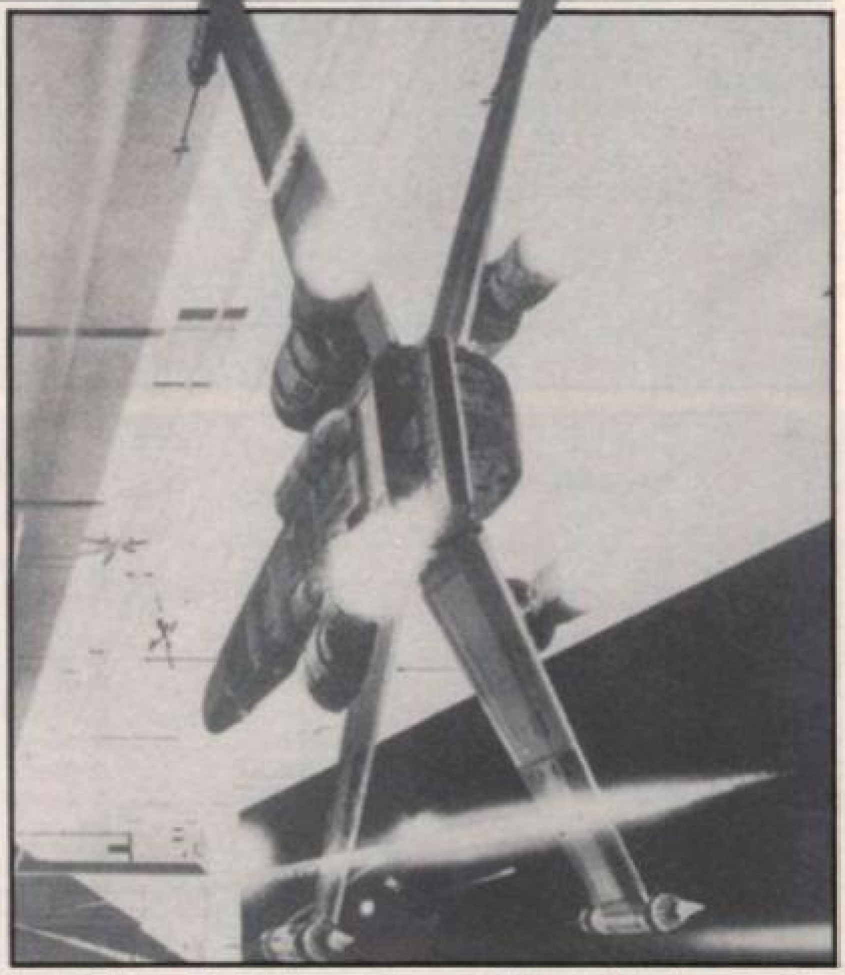
As it stands the program occupies ¼ KByte which is normally reserved for the printer so as to conserve memory.

The program is in six parts: the first contains two subroutines which can produce white noise (for gunshots and explosions) or tones (for sirens and zaps). The second, third and fourth parts produce the pre-defined sounds. The two remaining routines play the user-defined sound: the first with white noise (a kind of rushing sound) and the other with tones.

User defined sounds

The BEEP command produces a neat regular waveform reminiscent of an electric organ which is not very useful to the games programer.

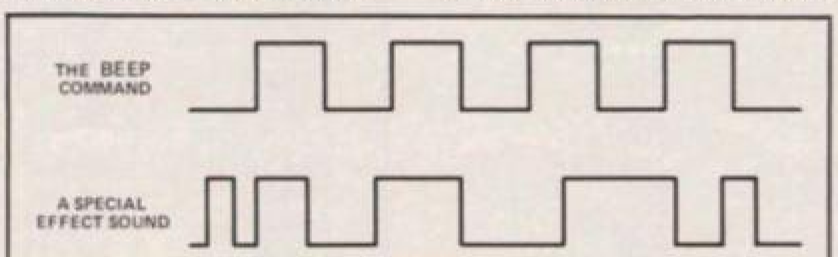
Special effect sounds are not so simple and require more information in order to be produced. With a BEEP command it is necessary to provide only two figures or parameters: the



first indicates how long you want the sound to last for and the second indicates the pitch of the resultant sound. A special effect sound can best be thought of as a succession of BEEPs one after another:

are to be produced, successive BEEPs will not suffice. Such sounds are produced with white noise which on its own gives a rushing noise.

Now to the practicalities: THERE IS NO NEED FOR THE



However, special effect sounds cannot be produced by the BASIC BEEP command because the interval between individual BEEPs is too great.

This, thankfully, is not an insurmountable problem, because by using machine code routines the delay between BEEPs becomes neglible.

If sounds such as explosions

USER TO KNOW ANYTHING ABOUT MACHINE CODE. All that is required is to follow these instructions:

1. Type in listing one.
2. RUN it.
3. Type SAVE ''Sounds''CODE 23296,256 & enter.
4. Rewind the tape and VERIFY''''CODE & enter.
5. If you get the message "Tape loading error" then go to stage 3.
6. The BASIC program has now done its job and it is safe to NEW. To recover the machine code routine use: LOAD '''' CODE.

If you have an assembler you should type in listing 2, assemble it and SAVE the code.

The next thing you will want to do is to listen to your new sounds. To do this simply enter the following:

```
10  PAUSE 0: REM waits for keypress
20  LET (an unused variable)=USR 23375; REM make a ZAP
30  GO TO 10: REM do it again
```

When this is RUN it turns your Spectrum into a ray gun. By changing line 20 to:

```
20  LET (an unused variable)=USR 23386
```

Now your Spectrum is a machine gun. And making line 20 read:

```
20  LET (an unused variable)=USR 23398
```

Now whenever you press a key your Spectrum happily explodes!

So to sum up:

LET (an unused variable) =USR 23375 produces a Zap.
LET (an unused variable) =USR 23386 produces a machine gun shot.
LET (an unused variable) =USR 23398 produces an explosion.

The more skilfull will want more than just these three and impressive sounds can be custom made. However, take great care: if you go wrong on this section you might have to wait several days before your machine has finished your sound.

So, to produce your own effects you must tell the computer the length and pitch of the eight beeps which make up the sound you want to produce. You must also tell it how many times to play each beep. Thus, for each beep there are three parameters: length, pitch and the number of times you want each beep to be produc-

ed. So in total there are 24 parameters. The procedure to tell these to the computer is:

```
10   DATA (length of 1st),
     (pitch of 1st),(No. of
     times), (length of
     2nd),(pitch of 2nd),
     (No. of times).... up to
     .... (length of
     8th),(pitch of 8th),
     (No. of times)
20   FOR a=23300 TO
     23323:READ d:POKE
     a,d: NEXT a
```

The length of each beep is a number from 1 to 255 and the longest note is 0.
The pitch of each beep is a number from 1 to 255 where 1 is the highest pitch (but 0 is the lowest).
Similarly the multiplier (how many times) goes from 1 to 255 (again 0 gives 256 cycles).

This program as it stands resides in the printer buffer which for most people is unused. However, if you enter any of the printer commands: LLIST, LPRINT or COPY then the sound will generate a delay followed by NEW, so take care.

Once you have provided the machine with the data you will need to know how to play the sounds. For this there are two commands:

```
LET  (an unused variable)
     =USR 23410
     this will play your
     sound with tones
LET  (an unused vari-
     able)=USR 23432
     this will play your
     sound with white
     noise.
```

Another slight snag encountered with this routine is that when the sounds are produced the border goes white. If you use an assembler to write in this program you can change it to suit your needs.

Hints

It is a wise idea to make the sounds last for as short as possible because it is then less obvious that the program stops to produce the sound. Also, the volume of the sounds is increased by resting the machine on a hard flat surface such as a desk or table.

Listing one (BASIC)

```
10  DATA 197,213,67,16,254,10,203,199,203,
    207,203,215,211,254,12,21,32,240,209,193,201
20  DATA 197,213,175,203,199,203,207,203,215,
    67,16,254,203,231,211,254,67,16,254,203,167,211,
    254,21,32,239,209,193,201
30  DATA 22,3,30,128,205,50,91,29,32,250,201
40  DATA 30,0,22,32,205,29,91,28,21,32,249,201
50  DATA 30,0,22,128,205,29,91,28,21,32,249,201
60  DATA 33,3,91,35,86,35,94,35,78,35,205,29,
    91,13,32,250,125,254,28,56,239,201
70  DATA 33,3,91,35,86,35,94,35,78,35,205,50,91,
    13,32,250,125,254,28,56,239,201
80  FOR a=23325 to 23453:READ d:POKE a,d:NEXT a
```

Listing for use with an assembler

ADDRESS	NMONICS	DECIMAL CODES
23325	push bc	197
23326	push de	213
23327	ld b,e	67
23328	djnz −2	16,254
23330	ld a,(bc)	10
23331	set 0,a	203,199
23333	set 1,a	203,207
23335	set 2,a	203,215
23337	out(254),a	211,254
23339	inc c	12
23340	dec d	21
23341	jr nz −16	32,240
23343	pop de	209
23344	pop bc	193
23345	ret	201
23345	push bc	197
23347	push de	213
23348	xor a	175
23349	set 0,a	203,199
23351	set 1,a	203,207
23353	set 2,a	203,215
23355	ld b,c	67
23356	djnz −2	16,254
23358	set 4,a	203,231
23360	out(254),a	211,254
23362	ld b,e	67
23363	djnz −2	16,254
23365	res 4,a	203,167
23367	out(254),a	211,254
23369	dec d	21
23370	jr nz −17	32,239
23372	pop de	209
23373	pop bc	193
23374	ret	201
23375	ld d,3	22,3
23377	ld e,128	30,128
23379	call 23346	205,50,91
23382	dec e	29
23383	jr nz −6	32,250
23385	ret	201
23386	ld e,0	30,0
23388	ld d,32	22,32
23390	call 23326	205,29,91
23393	inc e	28
23394	dec d	21
23395	jr nz −7	32,249
23397	ret	201
23398	ld e,0	30,0
23400	ld d,128	22,128
23402	call 23326	205,29,91
23405	inc e	28
23406	dec d	21
23407	jr nz −7	32,249
23409	ret	201
23410	ld hl,23299	33,3,91
23413	inc hl	35
23414	ld d,(hl)	86
23415	inc hl	35
23416	ld e,(hl)	94
23417	inc hl	35
23418	ld c,(hl)	78
23419	inc hl	35
23420	call 23326	205,91
23423	dec c	13
23424	jr nz −6	32,250
23426	ld a,l	125
23427	cp 28	254,28
23429	jr c −17	56,239
23431	ret	201
23432	ld hl,23299	33,3,91
23435	inc hl	35
23436	ld d,(hl)	86
23437	inc hl	35
23438	ld e,(hl)	94
23439	inc hl	35
23440	ld c,(hl)	78
23441	inc hl	35
23442	call 23346	205,50,91
23445	dec c	13
23446	jr nz −6	32,250
23448	ld a,l	125
23449	cp 28	254,28
23451	jr c −17	56,239
23453	ret	201

Zip!

A shorter and less powerful program than zappo, but easy to use and very effective. From David Mold in Hertfordshire.

This is a m/c program which produces a much more interesting sound from the Spectrum speaker than its usual 'beep'. It can be called very easily from BASIC and the sound produced can be varied by using DEF FN. The program was written for 16K Spectrum and could easily be run on a 48K model, although the addresses should really be changed to put it higher in memory on the 48K version.

The listing of the m/c is in three columns: the first shows the address of the corresponding bytes, the second shows the decimal codes of those bytes and the third shows the Z80 opcodes.

It should be entered to the address 32500 using this short program:

```
10     CLEAR 32499
20     DATA 42,11,92,17,4,0,25,78,25,25,70,58,
       72,92,203,47
30     DATA 203,47,203,47,238,16,211,254,81,
       21,32,-3,13,16,-11,201
40     FOR N=1 to 32: READ a
50     POKE 32499+n,a: NEXT n
```

The code can then be SAVEd using:

SAVE "beep" CODE 32500,32

and when it is required for use with a program — existing or to be written, it can be loaded from tape thus:

CLEAR 32499:LOAD "beep" CODE

and then used in conjunction with the program as described previously.

To call this routine from a program, first

DEF FN b(x,y) = USR 32500

then, when the beep is required use

LET 1 = FN b(N,N)

where N,N represents two numbers you can specify which, very broadly, set the duration & pitch respectively. They should both be between 0 & 255 inclusive. Experiment to find the effect of varying the two factors. Also, try

POKE 32528,12

as a direct command, and experiment further to find its effect.

Beep Routine

```
 32500   42,11,92     ld hl, (DEFADD)
 32503   17,4,0       ld de,4
 32506   25           add hl,de
 32507   78           ld c,(hl)
 32508   25           add hl,de
 32509   25           add hl,de
 32510   70           ld b,(hl)
 32511   58,72,92     ld a,(BORDCR)
 32514   203,47       sra a
 32516   203,47       sra a
 32518   203,47       sra a
*32520   238,16       xor 16
 32522   211,254      out (254),a
 32524   81           ld d,c
@32525   21           dec d
 32526   32,-3        jrnz @
 32528   13           dec c
 32529   16,-11       djnz,*
 32531   201          ret
```

Life and death

It's the meaning of life explained by John Elliot from Gloucester.

As many people who own a computer will know, Life is a simple algorithm for producing evolving patterns of cells. A grid is seeded with cells and the computer calculates and displays the growth of the colony by following simple rules; any cell which has two or three neighbours will survive, all others will die; any space which has three live neighbours will contain a live cell in the next generation.

This surprisingly simple set of rules can result in amazingly complex and beautiful patterns. These may evolve continuously, die away or become static, depending on the starting conditions.

Spectrum owners will have received a cassette from Sinclair which includes a version of Life. This is a very simple version which allows a simple programmed start colony which is then evolved continuously. Though this program does domonstrate the idea, it has a number of shortcomings which detract from the possibilities of the algorithm. In order to show the real appeal of Life, an implementation should show the following features:

1 It should be possible to have a random starting colony.
2 With a square grid, the colony breaks up when it reaches the edge. It should therefore provide a "wrapped round" screen which allows the pattern to 'fall' off one edge but reappear on the opposite edge. In this way moving patterns can be seen.
3 It should be possible to exit from the routine to change the positions of cells when required and then to re-enter and continue the evolution.
4 The routine must be fast so that the screen display does not become boring.

The Sinclair version is lacking in several of these respects so this program was written which incorporates these features.

Program RUNdown

The program was written in two sections; a BASIC controlling section and 408 bytes of machine code to carry out the calculations and handle the screen display.

The machine code is fully listed in Assembly for those interested, but the main sections are listed here in the order of execution rather than the order in memory.

Entering the program

First type in the loader shown in listing 2 and *save it!* Now run it (it takes about 20 seconds to load all the data into memory). If you then type RANDOMIZE USR 30351 the screen should clear and turn yellow. If not check your data lines. This is the CLEAR routine. Entering RANDOMIZE 30088 should cause a random generation of 'bugs' to appear and the colony should start to evolve. Press any key other than 1 and 2 and there should be no pause between generations. Pressing keys 1 or 2 will break out of the program at this stage. When you are sure the data is correct, resave it.

Now type in BASIC listing (listing 1) and save it as SAVE "life" LINE 1. Having done this, reload the machine code loader and rerun it. When the prompt appears this time, save directly after the BASIC program. You should now have the BASIC "life" on tape, followed by the code. Now load "life". It will run automatically, load the machine code and display the instructions.

This is the machine code program:

1 CLEAR (address 30351) This section sets up the screen by printing a whole screenful of 'bugs' in the same ink and paper colour. This has the effect of clearing the screen. A jump is then made to RAND for a random start or back to BASIC for a programmed start.
2 RAND (address 30097) This section steps through the ATTR file and, on the basis of a randomly generated number, changes the ink colour of about 30% of the positions so that the pre-printed bugs appear.
3 DUMP (address 30138) This then dumps the contents of the ATTR file into memory so that the new generation can be calculated.
4 PROG (address 30149) This is the main part of the program. It single steps through the dump, checking each position for the new generation and then loading the appropriate position in the ATTR file with either no cell (ink and paper the same) or cell (ink and paper different). The SOUND routine at 30398 is called at each cell position. This can be turned off by POKEing 30398,201 and turned back on by POKEing 30398,197.
5 KBDCHECK (address 30279) This causes a pause of about two seconds at the end of each generation and checks for a keypress. If key 1 has been pressed it returns to the BASIC menu. If key 2 has been pressed, control returns to the BASIC screen alter routine which allows you to change the composition of the colony. If there is no key press the program continues with the next generation. If any other key is pressed the program continues with no pause at all.
6 TABLE (address 30000) This table contains the data required to calculate the displacements within the dump of the eight positions surrounding each cell. It is this data which results in the 'wrap around' screen effect.

This is the small BASIC controlling program:

Lines 1-2	Clear space for the machine code and then load it into memory.
Lines 3-140	Set up screen colours and print instructions and menu.
Lines 150-160	Verifies input from keyboard.
Line 170	Runs machine code with random start.
Line 180	Terminates program.
Line 190	Runs CLEAR routine to set up screen for programmed start.
Lines 200-620	Prints cursor in the middle of screen and prompts for input. The cursor keys will move the cursor around the screen. Key 4 will enter a 'bug' at the current position. Key 9 will erase the bug at the current position. Key 3 will return to the machine code and continue evolving the colony. It is this section that control passes to from the machine code when key 2 is pressed.
Lines 1000-1010	These two subroutines set up the screen for printing to the bottom two lines. It is important not to break out while instructions are displayed here as this will cause a crash.

Listing 1

```
   1 CLEAR 29999
   2 LOAD "bytes1" CODE
   3 INK 9: PAPER 3: BORDER 4: C
LS
  10 PRINT  AT 2,8;"CONWAYS LIFE
"
  20 PRINT  OVER 1; AT 2,8;"____
_________"
  30 PRINT  AT 4,3;"THIS PROGRAM
 WILL SIMULATE"';" THE EVOLUTION
 OF A COLONY OF"';" ""BUGS"""
  40 PRINT  AT 8,3;"THE INITIAL
COLONY "'" MAY BE RANDOM OR PROG
RAMMED"
  50 PRINT  AT 11,3;"YOU MAY STO
P AT ANY TIME TO"'" CHANGE THE D
```

```
ISPLAY OR RETURN"'" TO THE MENU"
  60 PRINT  PAPER 2; INK 9; AT 1
7,3;"PRESS ANY KEY TO CONTINUE"
  70 IF  INKEY$ ="" THEN  GO TO
70
  80 BEEP .2,15: POKE 23659,2: C
LS
  90 PRINT  AT 1,13;"MENU"
 100 PRINT  OVER 1; AT 2,13;"___
_"
 110 PRINT  AT 5,3;"1:RANDOM STA
RT"
 120 PRINT  AT 7,3;"2:PROGRAMMED
 START"
 125 PRINT  AT 9,3;"3:END"
 130 PRINT  AT 12,2;"WHEN THE PR
OGRAM IS RUNNING,     KEY ""1"" W
ILL RETURN TO THE      MENU AND
KEY ""2"" WILL ALLOW     CHANGES
 TO BE MADE TO THE       SCREEN"
 140 PRINT  FLASH 1; INK 1; PAPE
R 9; AT 21,7;"ENTER YOUR CHOICE"
 150 LET a$= INKEY$
 160 IF  CODE a$<49 OR  CODE a$>
54 THEN  GO TO 150
 170 IF a$="1" THEN  GO SUB 1000
: GO SUB 1010: LET l= USR 30088
 180 IF a$="3" THEN  GO TO 700
 190 LET l= USR 30351
 200 BEEP .2,15: LET c=16: LET r
=11
 205 GO SUB 1000: POKE 23659,0:
PRINT  PAPER 4; INK 9; AT 22,0;"
use cursors to move,3 to return,
   4 to enter,9 to erase"
 210 DEF FN p()=22527+(32*r)+c
 220 LET p= FN p()
 230 LET z= PEEK p
 240 POKE p,18
 250 LET d= CODE  INKEY$ -48: IF
d<1 OR d>9 THEN  GO TO 250
 260 IF d=3 THEN  GO TO 400
 270 IF D=4 THEN  GO TO 500
 280 IF d=9 THEN  GO TO 600
 285 POKE p,z
 286 BEEP .1,10
 290 LET c=c+(d=8)-(d=5): LET r=
r+(d=6)-(d=7)
 310 IF c=33 THEN  LET c=1
 311 IF c=0 THEN  LET c=32
 312 IF r=22 THEN  LET r=0
 313 IF r=-1 THEN  LET r=21
 320 GO TO 220
 400 POKE 23659,2: POKE p,z
 405 GO SUB 1000: GO SUB 1010
 410 LET X= USR 30139
 500 LET z=48
 505 BEEP .2,20
 510 POKE p,170: GO TO 245
 600 LET z=54
 605 BEEP .2,0
 610 POKE p,170: GO TO 245
 700 POKE 23659,2: STOP
1000 POKE 23684,128: POKE 23685,
80: POKE 23686,33: POKE 23687,2:
 POKE 23659,0: RETURN
1010 PRINT  PAPER 4; INK 9; AT 2
2,0;"  enter 1 for menu,2 to alt
er                              "
1020 RETURN
```

Listing 2: The machine code loader

```
  10 DATA "DFFF01000100 1E0002001
E0001000100FFFF02001E00010001007
E0201000100E0FF01001E0002001E000
10001001E005FFD01000100 7E0201000
1001E000200C1FF1E00010001001E000
2001E000100"
  20 DATA "183C5A7E2424428121807
5227B5CCD8F762100580E160620E52A7
65CC5450E2909C10000007C22765CE1F
E3238043636180236302310E10D20DC0
1BE0211007D210058EDB011005821007
D0E1606203E16B928133E20B828143E0
1B92815B828"
  30 DATA "18DD2130751816DD21407
51810DD2150751 80ADD2160751804DD2
17075C5E5D53E3006080E00DD5E00DD2
3DD5600DD2319BE20010C10EFD1E179F
E032809FE0220127EFE30200D3E3012E
5216400CDBE76E1180B3E3612E521000
2CDBE76E1"
  40 DATA "1323C1108B0D20860E960
6FFFDCB016E200C10F80D20F3FDCB01A
ECABA7521085C7EFE312011218D5C361
821500022425C21445C3601C9FE32200
C21C80022425C21445C3601C90000FDC
B01AEC3BA75FDCB0286216B5C3602210
04022845C212118"
  50 DATA "22885C0000002100580E1
606203E90D736362310F80D20F3FDCB0
1AEC9C5D5110100CDB503D1C1C9"
  60 RESTORE
  70 CLEAR 29999: LET A=30000
  80 FOR B=1 TO 5
  90 READ C$
 100 LET D=(( CODE C$-48)-(7*( C
ODE C$>57)))*16
 110 LET E=D+(( CODE C$(2)-48)-(
7*( CODE C$(2)>57)))
 130 POKE A,E
 140 LET A=A+1: LET C$=C$(3 TO )
 150 IF C$>"" THEN  GO TO 100
 160 NEXT B
 170 SAVE "bytes1" CODE 30000,41
0
```

To date, Atari's most astonishing game is Pole Position... If you are only going to buy one game, then this is the one you should get.
COMPUTER & VIDEO GAMES.

...Pole Position gives a very strong sense of speed as you hurtle round the track. The super-realism of the three-dimensional effect adds a lot to the game. It is a great graphics demo.
PRACTICAL COMPUTING.

Brilliant!
WHICH MICRO? AND SOFTWARE REVIEW.

...a terrific version of the arcade motor racing game...graphics are superb...sound, too is very good as brakes screech and engines rev-up.
PERSONAL COMPUTER GAMES.

Exciting, exhilarating, excellent, Pole Position takes the lead as the best Atari race game around.
PERSONAL COMPUTER NEWS.

Escape from Omicron

Sean Morley of County Mayo challenges you to escape or die in this exciting ZX81 game.

The planet of Omicron, graveyard of many fearless space explorers, is about to claim another victim — YOU.

However, you have one last chance to survive — a shuttle has been sent to rescue you, all you have to do is reach it. It has landed at the opposite edge of a rocky plain that you will have to cross, so you must leave the safety of your cave and set off.

This would be an easy task were it is not for the savage Crag who will give chase as soon as he sees you. You must, of course, go around the rocks — inverse Xs (these are in line 20 of the listing and have been listed as "x") — but beware! Crag can chew his way through them!

To guide your man, "*", use the cursor keys 5,6,7 and 8. Good Luck.

Variables

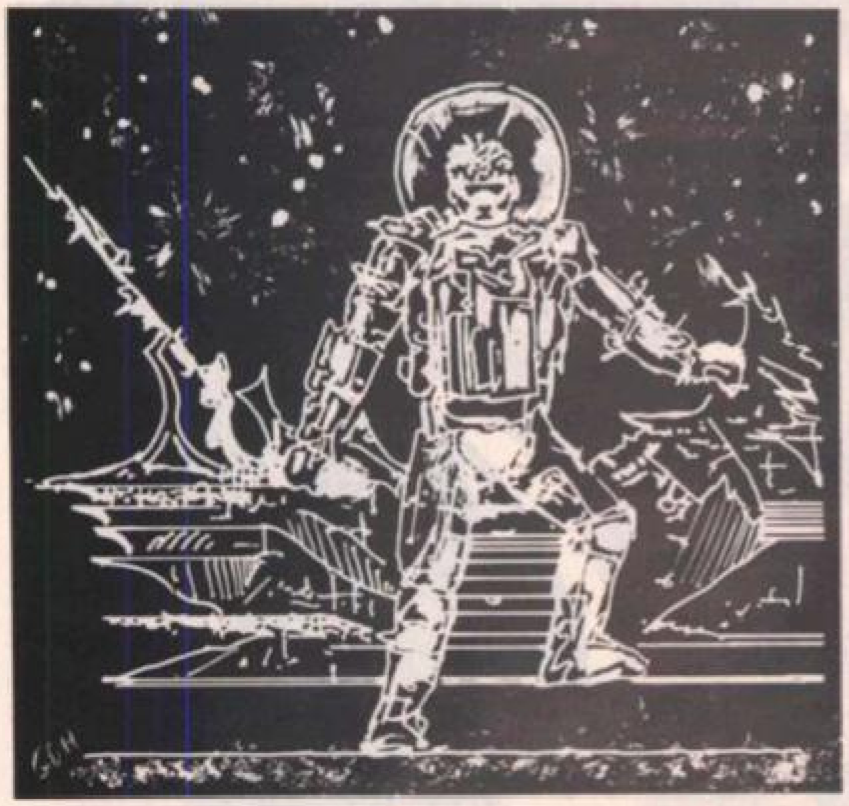
A	Crag's vertical position.
B	Crag's horizontal position.
X	Your vertical position.
Y	Your horizontal position.
L	Level of difficulty.
A$	Key depressed.

Graphics

In line 20 rocks are represented by inverse Xs.
In line 55 the shuttle is represented by a graphic Q and a graphic W.

How it runs

4	Goes to instruction routine.
5	Sets levels of difficulty.
6-30	Draws rocks at random positions.
31	Clears rocks printed around you before you start.
40-50	Initialises variables which hold your position.
55	Prints shuttle.
60-70	Initialises variables which hold Crag's position.
80	Tests for key depression.
81	Checks to see if you were eaten by the Crag.
82	Reduces Crag's movement depending on level of difficulty.
85-87	Calculates Crag's new position.
90-95	Calculates your new position.
100-110	PEEKS your position to make sure it's empty. If there is a rock there your move is cancelled.
120-130	Cancels your move from lines 100-110.
134	Erases you.
135	Erases Crag.
136	Prints you in original position after a cancelled move.
137	Returns to line 80.
140	Prints you in your new position.
150	Prints a space behind you depending on which way you moved.
153	Checks to see if the Crag got you.
155-156	Blanks out you and Crag.
160	Goes back to line 80.
1000-1010	Prints end routine after you have been eaten by the Crag.
1020-1030	Checks to see if Y or N was pressed for new game.
2000-2020	Prints end routine after you have reached the shuttle.
2030-2050	Checks to see if Y or N was pressed for new game.
4000-4020	Prints instructions.
4030-4055	Inputs level of difficulty and checks to see that it is between 3 and 10.
4065	Waits for 2 seconds.
4070	Waits for key to be pressed before continuing game.
4080	Clears the screen.
4090	Returns to game.



```
   4 GOSUB 4000
   5 LET L=L/10
   6 RAND
  10 FOR A=0 TO 100
  20 PRINT AT RND*21,RND*31;"x"
  30 NEXT A
  31 PRINT AT 0,1;" ";AT 1,0;" "
  40 LET X=0
  50 LET Y=0
  55 PRINT AT 20,30;CHR$ 130;CHR
$ 129;AT 19,30;" ";AT 20,29;" "
  60 LET A=20
  70 LET B=0
  80 LET A$=INKEY$
  81 IF A=X AND B=Y THEN GOSUB 1
000
  82 IF RND>L THEN GOTO 87
  85 LET A=A+(1 AND A<X)-(1 AND
A>X)
  86 LET B=B+(1 AND B<Y)-(1 AND
B>Y)
  87 PRINT AT A,B;"O"
  90 LET Y=Y+(1 AND A$="8" AND Y
<30)-(1 AND A$="5" AND Y>0)
  95 LET X=X+(1 AND A$="6" AND X
<20)-(1 AND A$="7" AND X>0)
  96 IF X=20 AND (Y=30 OR Y=31)
THEN GOSUB 2000
 100 PRINT AT X,Y;
 110 IF PEEK (PEEK 16398+256*PEE
K 16399)=0 OR PEEK (PEEK 16398+2
56*PEEK 16399)=52 THEN GOTO 140
 120 LET Y=Y-(1 AND A$="8")+(1 A
ND A$="5")
 130 LET X=X-(1 AND A$="6")+(1 A
ND A$="7")
 133 IF A=X AND B=Y THEN GOTO 10
00
 134 PRINT AT A,B;" "
 135 PRINT AT X,Y;" "
 136 PRINT AT X,Y;"*"
 137 GOTO 80
 140 PRINT "*"
 150 PRINT AT X,Y+1;(" " AND A$=
"5");AT X,Y-1;(" " AND A$="8");A
T X+1,Y;(" " AND A$="7");AT X-1,
Y;(" " AND A$="6")
 153 IF A=X AND B=Y THEN GOTO 10
00
 155 PRINT AT X,Y;" "
 156 PRINT AT A,B;" "
 160 GOTO 80
1000 CLS
1010 PRINT AT 8,0;"YOU WERE EATE
N BY THE CRAG.      FOR ANOTHER G
O HIT Y IF NOT HIT        N"
1020 IF INKEY$="Y" THEN RUN
1030 IF INKEY$="N" THEN STOP
1040 GOTO 1020
2000 CLS
2010 PRINT AT 8,0;"YOU MANAGED T
O REACH THE          SHUTTLE IN TI
ME.WELL DONE"
2020 PRINT AT 11,0;"FOR ANOTHER
GO HIT Y.IF NOT HIT  N"
2030 IF INKEY$="Y" THEN RUN
2040 IF INKEY$="N" THEN STOP
2050 GOTO 2030
4000 PRINT "       ESCAPE FROM OMI
CRON","     ====================
"
4010 PRINT AT 6,0;"YOU ARE STRAN
DED ON THE ROCKY   PLANET OF OMI
CRON.A SHUTTLE HAS BEEN SENT TO
RESCUE YOU BUT YOU HAVE TO CROSS
 A ROCKY PLAIN TO  REACH IT"
4020 PRINT AT 10,9;"BUT AS SOON
AS YOU       LEAVE YOUR CAVE THE S
AVAGE        CRAG CHASES YOU. HE I
S ABLE TO   EAT HIS WAY THROUGH
 ANYTHING"
4030 PRINT AT 16,0;"ENTER A LEVE
L OF DIFICULTY(1-10)"
4040 PRINT AT 17,0;"LEVEL 1 IS V
ERY EASY AND LEVEL  10 IS    VERY
 DIFFICULT"
4050 INPUT L
4055 PRINT AT 15,0;"USE ARROW KE
YS TO MOVE"
4060 PRINT AT 19,0;"HIT ANY KEY
TO BEGIN"
4065 PAUSE 100
4070 IF INKEY$="" THEN GOTO 4070
4080 CLS
4090 RETURN
6000 SAVE "ESCAPE FROM OMICRON"
6010 RUN
```


CLEVER CLOGS
EARLY LEARNING GAMES
It's ARRIVED!
— the next generation of
EARLY LEARNING SOFTWARE
Games that make fun out of learning
Hand-built by teachers, played by children
JUNGLE JUMBLE (5+)
WHIZZ QUIZ (7+)
STAR TRUCKER (9+)
PARTY TIME (3+)
SHIPSHAPES (3+)
SAM SAFETY (5+)
MUSIC (7+)
BLOCKBUSTER (7+)
Available for Spectrum, BBC and Commodore 64 computers
APS
SOFTWARE RETAILER
Available at W.H. Smith and other leading software outlets
FROM
Computertutor: A member of the Argus Press Software Group
No 1 Golden Square, London W1R 3AB.

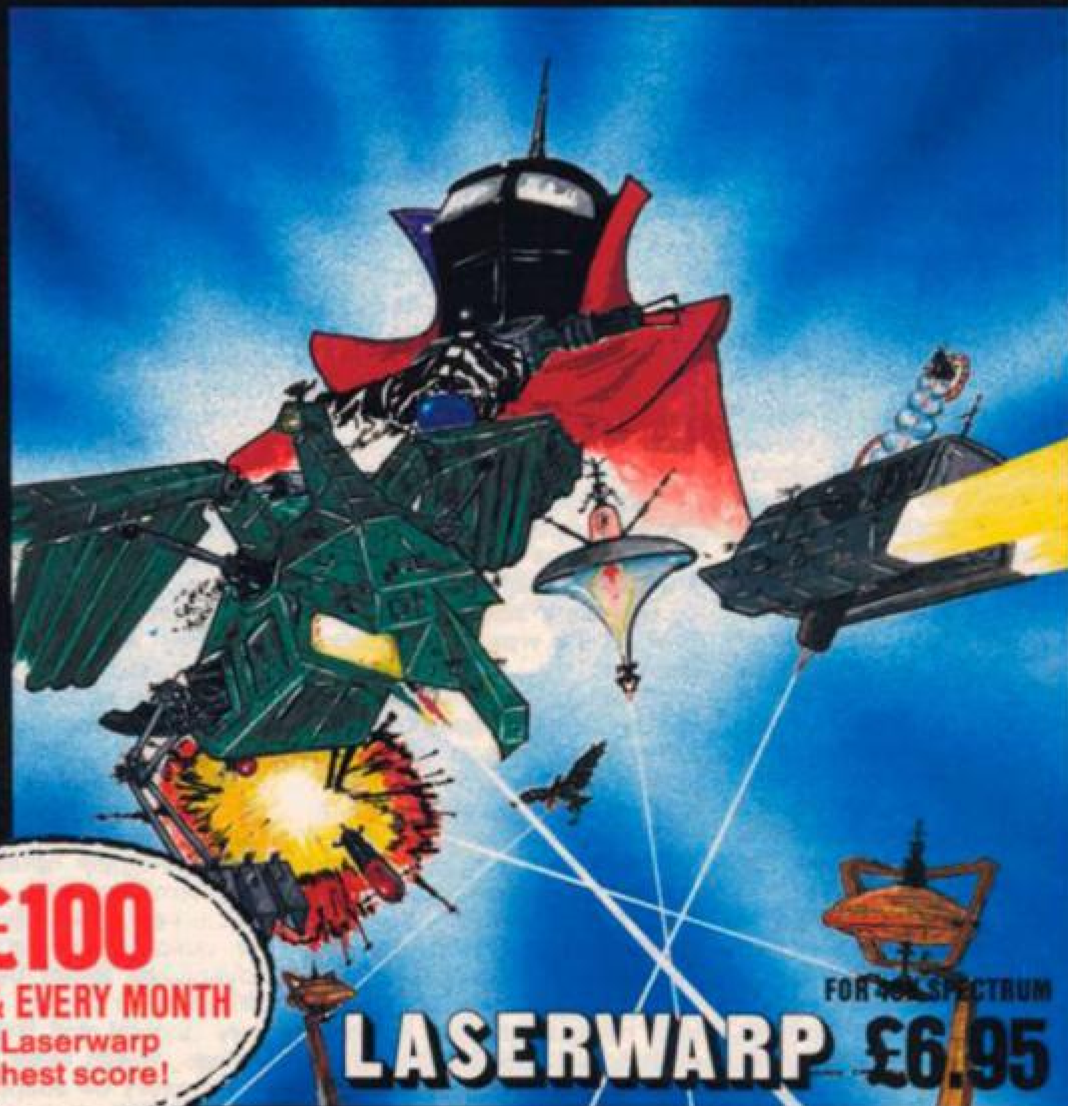
Can you meet Laserwarp's infinite Challenge?

Be a real game lord – imagine the quest for the ultimate high score!

At last – a real challenge for the arcade ace: Laserwarp, the most exciting arcade-style game around, in full, high-res graphics. It's got everything – fight your way through the ramships, space mines, homing droids, hyperspace chickens, interstellar pogos, and more, till you challenge the Master. Can you beat him? Can anybody beat him? Comes complete with Hall of Fame, coded verifiable high score – send yours in and see if you can win the £100 Mikro-Gen are offering every month (full details in instructions)

High scorers names will be published in a Hall of Fame in the computer press – and they will be invited to join in a Battle of the Champions.

NB – If you have the earlier version of Laserwarp, Mikro-Gen will replace it with the new version – FREE!



£100
EACH & EVERY MONTH
for Laserwarp highest score!

* Genesis II * *

the adventure game that understands plain English!

The colony ship Genesis II has been hijacked by space pirates – and you're the only crewman left! Only you can save the cargo – an entire forest, full of animals, destined to populate a new world. This exciting adventure game has a vocabulary of nearly 300 words, avoiding that frustrating search for the precise word command.

- ★ Nearly 100 different locations
- ★ Over 60 objects ★ 100% machine code
- ★ Over 500 possible actions

* For 48K Spectrum
£6.95



Just part of Mikro-Gen's great range – look out for Paradroids, Mad Martha II, Cruise Attack, Land of Sagan, Galakzions, Deffendar and many others. Available from WH Smith, Co-op, Boots, Menzies & other leading retailers, or direct from Mikro-Gen (Please make cheque/PO payable to Mikro-Gen & add 40p post & packing

PHONE YOUR VISA/ACCESS NUMBER

MIKRO-GEN

44 The Broadway, Bracknell, Berks RG12 1AG Tel: 0344 427317

Bookshelf

Patrick Cain gives us his thoughts on some of the latest books available for the Sinclair user.

Using Your Home Computer — Garth W P Davies

It was an enquiry from a reader that brought this book to my attention: there aren't many books available for practical computer applications in the home. A short time later I came across Using Your Home Computer: the subtitle, Practical Projects for the Micro Owner, suggested that this might be a book that could fill the gap. Both Mr Davies and the publishers, MacMillan Press, might well be on to a good thing here, I couldn't wait to find out what uses all we frustrated serious applications people could be putting our minds to around the home.

On the principle that if you give a man a fish you feed him for a day but teach a man to fish and you feed him forever — Garth Davies goes further than the titles suggest and attempts to lay guidelines and establish a good working procedure for designing a variety of practical projects. To do this he can draw on his extensive commercial background in systems analysis. In fact, if this book teaches anything then it is how to set about analysing problems. The theory then is that having acquired a method of breaking down possible tasks to identify the requirements, the solving will be so much simpler and all that really is needed to make full use of your home computer is a handful of clever ideas. Without any doubt this analytical approach has proved itself well in business.

Fortunately, the text does not burden readers with theoretical planning models or jargon terms of business use, but discusses each topic with reference to realistic home examples. By adopting this method the author is able to explain the logic and highlight pitfalls with tangible examples. The result is a well written text that clearly explains the technique. Although I do not think a Christmas card list compiler (the example used in the discussion) is a terrific idea, it is an effective example.

Later chapters offer a further selection of projects, in fact, a total of 23 topics range from a diet plan, recipes and menus, football pools forecasting to examination questions and mortgage repayments. Each project specification follows the theory already discussed. Readers who closely follow the specifications in their own projects will find themselves with the largest part of the work already done and well on the way to completing the rest.

At no point in Using Your Home Computer does any computer code appear: all project suggested could be applied to any machine, given that sufficient memory is available. It follows then that readers are expected to be capable of coding from the specifications arrived at, but that should not prove too difficult. The main difficulty will lie in finding worthwhile uses for your home computer. I am not entirely convinced that many of the twenty four examples included within the paperback's eighty pages fall into that category.

Whether this book will be a real aid to you will depend on the ideas you have for using your computer around the home. There is little doubt that a practical method for examining your ideas and analysing the problem to produce a workable project specification can be learned, professional systems analysts will confirm that, this book can teach these things well. Perhaps some further examination would show that few things in the domestic environment require such effort.

Using Your Home Computer is written by Garth W P Davies and published by MacMillan Press. ISB 0 333 352173.

Creating Adventure Programs on Your Computer — Andrew Nelson

Have you noticed the new brand of superhuman that is around these days? When rising unemployment, high cholestrol levels and the likelihood of overexposure to the Olympic Games are enough for most of us, there are those of rare good sense, who, without regard to witches, ogres, trolls or their brains seek excitement in the world of adventure games. Now for those who have been gifted with more than their fair share of superhuman powers, there is Creating Adventure Programs on Your Computer, an Interface Publication written by Andrew Nelson: Superheroes read on.

The book itself is quite an adventure. At times I had to rely on my own superhuman skills to fathom it out. An exciting battle with the binder notes followed by skirmishes with the pages that lurked within failed to reveal for which particular computer adventures might be created, although, to be fair, the text is very general and the program listings, which were in BASIC, could easily be adapted to suit any BASIC dialect. Equally mysterious was the price; although I'm sure there must be one, there was no evidence of such, which makes it a bit tricky to judge whether the soft back and 236 pages between represent good value.

Adventure games represent a substantial proportion of the software available for home micros, while books on their creation are something of a rarity. It was refreshing to discover a text removed from the run of the mill '100 Imitations Of Your Favourite Arcade Games' type that are in such great abundance. Having recognised some uncharted or unwritten for territory, has Andrew Nelson been able to set his mark on the ground he pioneers? Will the reader, as promised on the binder notes, "soon be creating magnificent Adventure Programs"; Without leading you a merry dance or on a wild goose adventure the short answer is No. Well, maybe.

If the type of adventure you seek is the picture sort, then go no further for this book is not intended to offer guidance on graphics. Andrew Nelson's adventures are word based and rely on you to draw your own pictures. That is no bad thing for it means that the scope of the adventure can be greater than one that was restricted by graphics. It also means that the design of the game is significantly more simple and can easily be discussed in full within the book.

While any adventure game will be judged for its individuality, it is the effectiveness of its design structure — common to all adventure games — that makes it reliable. In the eighteen months that Andrew Nelson has spent 'playing, devising and programming' adventure games he has learned well the importance of structure and devised his own effective method which he introduces in the book. The three adventure programs contained, 'Werewolves and Wanderer', 'The Aftermath of the Asmovian Disaster' and 'The Citadel of Pershu' are examples of the application of his design. The text uses these to show how from an initial idea a full adventure can be built. From laying the floor plan to developing handling techniques to a final elaboration of the game, reference is made to one of these examples. By following the author's guidelines, a systematic approach to adventure games can be developed and applied to your own efforts. No book can guarantee that it will enable you to create 'Magnificent Adventure Games', but many abstract concepts have been explained and once attained the reader is empowered to create a reliable adventure base.

Unfortunately, while it was possible to recognise and appreciate Nelson's technique and indeed to learn from it well for it is not too demanding or complicated, it was difficult to follow his train of thought and I found that several readings and a fair deal of intuitive guess work were required. "All I can do is ask you to proceed on trust" is to me a rather lame request, better perhaps if the author had spent a few more months planning the book as carefully as he

had studied adventure games. I wonder if the publishers were aware of the deficiency and the need for repeated reading and made allowances for such by including pages 189-302 twice.)

Achieving magnificence is an individual thing. A magnificent adventure program depends on a reliable structure and the inventiveness of the creator; that inventiveness depends on the creator's background knowledge. Chapter 21, 'Adding Excitement', offers several pointers to features of a good game and sources of information and background. If the reader applies the technique offered by the author, he can create an adventure game; by heeding the advice on characteristics of good games, the reader will be in a position to create a better program. These are the things that this book can teach. What a pity the text is at times unclear for all the right information is included. Programmers with a working command of BASIC will understand any programming techniques used, maybe only those with adventure experience will succeed in unravelling the text.

Creating Adventure Programs On Your Computer by Andrew Nelson is priced at £4.95 and published by Interface. ISBN 0907563 36 8:



First steps in BASIC — Susan Curran and Roy Curnow

The credits of husband and wife team, Susan Curran and Roy Curnow read like the glittering prizes of the commercial and personal computing world. The computer titles they have jointly and individually been responsible for total far too many to include here. Susan is a specialist in User Applications of Micro Computers as well as a director of a consultancy company. Roy squeezes being a Professor at City University, London in alongside with being an advisor to several multinationals, an I T Consultant, an author and a company director! He was also the man the BBC made responsible for structuring and developing the 'Computer Program' television series. Their pedigree alone makes their current book, 'First Steps in Basic', a Windward publication, well worthy of some attention.

As any new programmer will agonisingly testify, some computer manuals are precise to the point of being sinful; others are simply muddled or remiss. I well remember my own frustrated attempts at mastering this 'genius level' code and recognising the need for a more patient set of instructions. It is, after all, the manual's job to detail the features of the language as applied to the machine and not to teach programming. Of course, many more resourceful people than I have recognised this need, resulting in a crock of books of varying standard, aimed at the novice computer programmer. As one might expect, this one stands out from the crowd.

It is fair criticism, I think, to say that it might have been better if the authors had chosen to devote the book to one type of computer entirely. Instead, a non-specific text using 'Micro soft BASIC' as an example caters for most computer types. An appendix details how the dialect used varies from that run on other machines, including the Spectrum. But far outweighing any criticism must be the credit due to a well planned and thoughtful book that is aware of those 'early days' pitfalls.

The reader is not expected to have any previous knowledge of computing and no more is required of him than that he has connected the various wires and switched on. In fact, allowances have been made for readers who may as yet not be in possession of a computer, although why anyone should want to learn to program without one is beyond me. It is my opinion that a lot of heads are banged against walls as novice programmers toil with a whole new confusing world, unable to identify and recognise the many constituent parts. An introductory chapter that outlines the computer, its requirements and functions, the basic language — the reason for it and its applications and more key elements of computing should help prevent future damage to both heads and walls.

Chapter One, 'Writing on the Screen' deals with the PRINT statement in its various forms, writing messages on the screen, positioning screen output and simple arithmetic; introducing and explaining on the way strings expressions, arguments and variables. Each topic is discussed in straightforward text, highlighting each point or instruction with an example and supporting screen illustration. Not, of course, a new approach to this subject but one that works well because the authors have identified likely areas of difficulty and developed the text accordingly. (NB: page 17: don't get too worried if your computer gives a different answer to the cosine of 43, unless the laws of mathematics have changed this is a printers error: at least that's my angle.)

Subsequent chapters build on the knowledge and expertise that has been built up in the first few pages. By chapter 2 this expertise is put to good use in writing short programs, reinforcing the theory learned so far while introducing new elements of the computer's ability, eg 'loading' and 'saving'. As throughout, the text is straightforward and unassuming, expanding and elaborating ideas to give a fuller awareness of the topic. Supporting examples are fun and interesting enough to try, while again clear illustrations compliment and highlight the text.

Further chapters take task in 'loops and branches', 'editing and debugging' and 'handling data' until a comprehensive knowledge of BASIC statements and familiarity with their applications is gained. Chapter 7, 'Writing Longer Programs', brings these statements together and sets guidelines for planning and writing more involved programs. On conclusion the reader should be familiar with BASIC statements and aware of good program structure. At this point it is necessary to develop the skills learned, a job that is beyond the scope of this book, and readers are directed to further reading material. New programmers will find that in investigation of subsequent texts they have a firm foundation as a starting point for develping their programming skills.

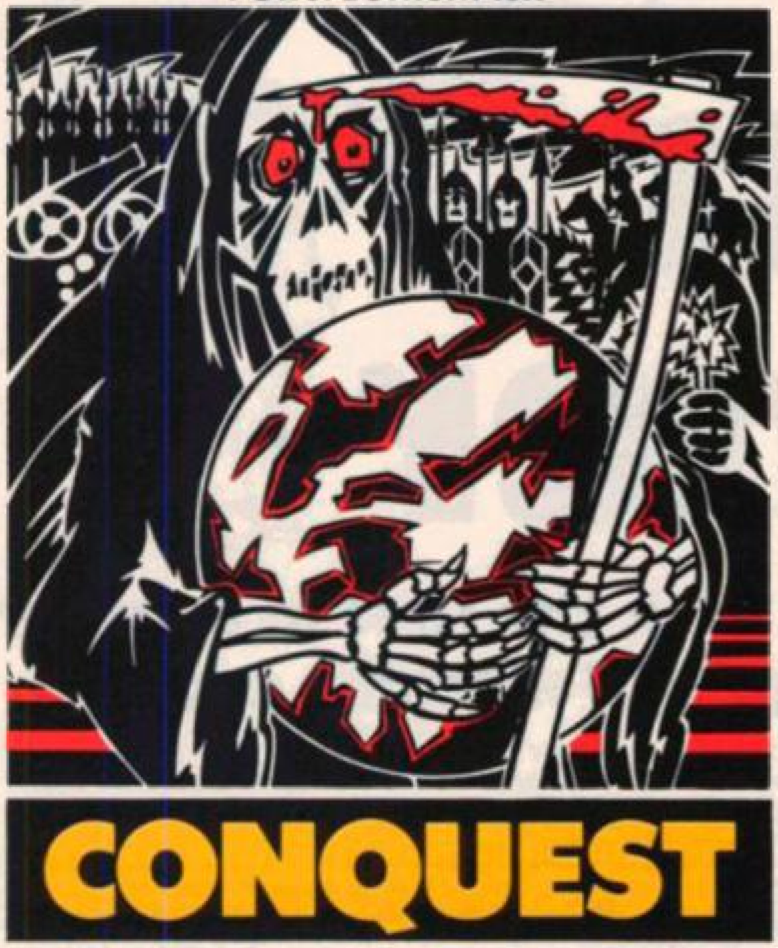
First Steps In BASIC by Susan Curran and Roy Curnow is part of the Clear and Simple Home Computing Series published by Windward and costs £4.95. ISBN 0 7112 0333 4



ONLY ACES SURVIVE IN
NIGHT GUNNER
48K SPECTRUM
BY R.J.SWIFT
JOYSTICK COMPATIBLE (Kempston, AGF, Sinclair, Protek and Others)
£6.95
– THE AMAZING AIR BATTLE SCENARIO THAT STRETCHES YOU TO THE VERY LIMIT – GUNS BLAZING, ENEMY FIGHTERS ATTACK FROM ALL DIRECTIONS – 3D GROUND ATTACK SORTIES AND THERE'S STILL THE FLIGHT BACK HOME.
OTHER AMAZING FEATURES
• 30 GROUND ATTACK MISSIONS
• REALISTIC 3D EFFECTS
• EXPLOSIVE SOUND
• 1 OR 2 PLAYERS EACH WITH 4 LEVELS OF DIFFICULTY
• HIGH SCORE TABLE WITH MEDALS
• SELF DEMONSTRATING TITLE PAGE
Selected products available from:
Boots, Woolworths, H.M.V., Menzies, Greens, Lasky's, Rumbelows, Selfridge, Spectrum Dealer, Video Palace, Zapp and all good Computer shops.
TAXI
£5.50
Any Spectrum
Taxi
Become a Taxi Driver and test your skills – but beware the Reckless Drivers!
Fun for all the family.
FOOTBALL LEAGUE ANALYSIS
48k Spectrum
FOOTBALL
£7.95
48K Spectrum
League Analysis – Easy to use Pools Prediction.
With 9 methods of team performance analysis – let your Spectrum guide you to your pools selection.
FIGHTER PILOT
48K SPECTRUM
FIGHTER PILOT
£7.95 48K Spectrum
Joystick compatible.
A spectacular flight simulation of the world's most exciting jet fighter with stunning 3-D cockpit view and the most realistic air to air combat ever seen on the spectrum!
DIGITAL INTEGRATION
Watchmoor Trade Centre, Watchmoor Road, Camberley, Surrey GU15 3AJ
Please send me the following games for the Spectrum
Night Gunner 48K £6.95
Fighter Pilot 48K £7.95
Football 48K £7.95
Taxi (Any Spectrum) £5.50
VAT and p. & p. inclusive within UK (Overseas inc. 55p. per cassette)
Trade and Export enquiries welcome.
Lifetime Guarantee Cheques payable to Digital Integration
I enclose a cheque/P.O. for ______ Total.
Name ______
Address ______
Or debit my Access/ Account No.
Tele Sales Camberley (0276) 684959
Please send to Digital Integration, Dept
Watchmoor Trade Centre, Watchmoor Road, Camberley, Surrey GU15 3AJ

CheetahSoft
Soft we're not

FOR SPECTRUM 48K



CONQUEST A tactical game which even veteran players will find both challenging and rewarding.

Mediterranean Europe is divided into grid squares. Your aim, as Emperor, is to gain 100 squares of territory as quickly as possible – at the same time dealing with Barbarian counter-attacks, plagues, civil war and rival Emperors.

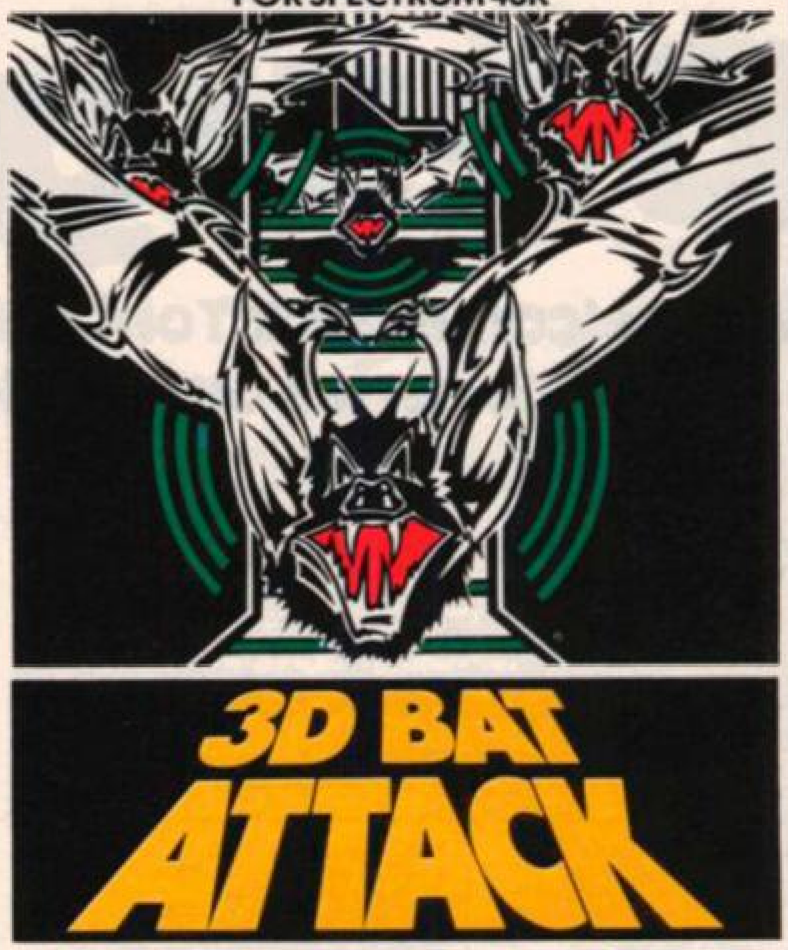
5 levels of skill, plus a continuous 'Conquest' game where all the difficulty levels are thrown in together.

3D BAT ATTACK An all action, 3 dimensional maze game where you gather up blocks of gold, at the same time pitting your wits against vicious vampire bats whose only purpose in life is to locate, hunt and kill you.

4 levels of skill. At each level the game gets faster and more complicated, and the vampires more dangerous.

CheetahSoft
Soft we're not

FOR SPECTRUM 48K



AFTER THESE, THE REST IS KIDS STUFF.

Are you ready for CheetahSoft? There's only one way to find out. But be warned: these vampire bats know a good meal when they see one. And our friend with the scythe has had years of experience . . .

So don't play unless you're ready to play the game for real. Because you'll find there's one sure thing about CheetahSoft: Soft we're not.

£6.95 at all good stores.

CheetahSoft
Soft we're not

Light Screen Designer



We welcome back Toni Baker with the first in another superb series on the delights of machine code.

Well, hello everyone, and welcome to the first in a brand new series of articles. Throughout this series I intend to list and explain one program — only one, mind — not one per issue but one program altogether throughout the series! Why? Well it's a bit long, that's all.

One problem you get with long programs, I've found, is that lots of errors seem to crop up in the listing, so we'll have to see what we can do about that. For those of you who've got the word processor program, WordSheep, from the Feb/Mar issue, but haven't managed to get it going yet, here are the corrections you need to make:

Page 45:	In COMPRESS:	ED80 should read EDB0
	In ADJUST:	010060 should read 016000
Page 46:	In A__RANDOM:	20FD should read 30FD
	In TRANSFER__U:	013080 should read 010380
	In WIPE:	23 should read 13
Page 47:	In DELETE	MISSING INSTRUCTION: C9 (RET) from end of subroutine.
Page 48:	IN PROCESS:	2100EA should read 21C0EA
Page 49:	In REFORM:	MISSING INSTRUCTION: 09 (ADD HL,BC) between 2AAF5C and 012000

Anyway — back to the present. Light Screen Designer is for people who've got 48K knocking around inside their Spectrum. If you've only got 16K then bad luck — you can't use this program — not because it isn't relocatable, but because you just haven't got enough memory — this really is a long program!

It's a program to help you design pictures on the screen, a grand artwork program incorporating just about every feature you could possibly desire to help you draw pictures. You can do straight lines and curves and various geometrical shapes. In fact the program will even do the colouring in for you once you've drawn the outline — in any colour too!

If you hunt around these pages you'll soon come across a picture labelled Fig. 1. You have to MAKE one of these. It's a keyboard overlay and tells you what each of the keys do when the program is running. To make an overlay, get a piece of card the same size as a Spectrum keyboard, cut holes out for the keys to go through, and write all the appropriate words above the holes. Alternatively, if you can't be bothered to do all that, just keep a copy of Fig. 1 handy by your Spectrum when running the program and refer to it as and when you need to. I shall explain what each key actually does later on, when we actually get down to writing the individual parts.

Let's get down to it then shall we? The memory used by the program starts at address C000, although the program itself doesn't actually start until we reach address DB42, so here's what's what:

C000-D7FF	DFC	Display File Copy — used to record pictures.
D800-DAFF	AFC	Attributes File Copy — used to record pictures.
DB00-DB41		System Variables used by the program.

You don't have to worry about *any* of this for the time being. Now for some work. The addresses from DB42 to DB8F store a table of subroutine addresses. The table will be called CMD__ADDRS. Since *no* subroutines exist at present then you should, for the time being, set every entry in this table to 0052 (an address in the ROM at which a RET instruction is stored). To do this you must POKE 52h into address DB42, 00 into DB43, 52h into DB44, 00 into DB45,... and so on up to ...,52h into DB8E and 00 into DB8F. Got all that? Right, now for some more data — a table called INP__TABLE. Feed this in:

```
                              ORG DB90
59 CE 38 31    INP__TABLE     DEFM "Y","N *","8","1"
B0 0D 39 38                   DEFM "0 *","enter","9","8"
37 36 35 34                   DEFM "7","6","5","4"
33 32 31 B0                   DEFM "3","2","1","0 *"
```

It won't make a lot of sense to you at present, but all will become clear in a moment, so don't go away. Note that some of the bytes have 80h added, and so I've marked these with an asterisk in the right hand column. Some more data coming up next, but in a slightly different form. What we have are eighteen messages which will later need to be printed at the bottom of the screen.

Each message is followed by a coded byte which will tell the program if anything needs to be input. To enter this, note that the first column contains the address to which the message is to be written, the second column contains the text of the message which must be loaded character by character from the specified address, and the third column contains one byte of data which must be POKEd immediately after the text of the message (note that this data is given in *hex*).

ADDRESS	TEXT	DATA BYTE
DBA0	Light Screen Designer	80
DBB6	Paint Colour	96
DBC3	WARNING — Screen Memory Wipe	90
DBE0	Store Memory	95
DBED	Recall Memory	95
DBFB	Clear Screen	90
DC08	Number of Radians? *space*	80
DC1C	Now in Text Mode	80

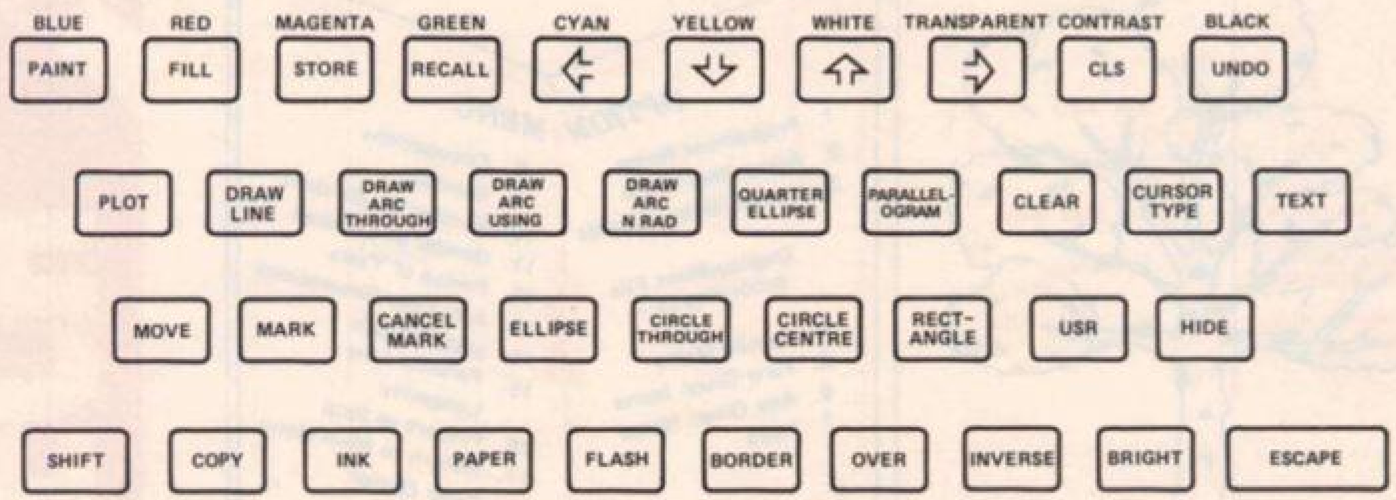


Fig. 1

```
DC2D    Machine Code Address? space   80
DC44    Copy                          90
DC49    Ink Colour                    96
DC54    Paper Colour                  96
DC61    Flash Status                  92
DC6E    Border Colour                 98
DC7C    Over Status                   9E
DC88    Inverse Status                9E
DC97    Bright Status                 92
DCA5    Return to BASIC               90
```

Now so far I haven't really given you anything useful except a lot of promises for the future, so I'd now like to implement just two subroutines in the program. The techniques involved are quite clever, although not difficult to follow, so I imagine you should be able to learn from them. Firstly, we have a subroutine called GET_CHR which merely waits for any key other than "CAPS SHIFT" to be pressed, and then returns with DE containing a current keyboard scan.

```
                     ORG DCB5
CD8E02  GET_CHR      CALL 028E,KEY_SCAN   DE: = keyboard scan.
7B                   LD A,E
FE27                 CP 27
2803                 JR Z,GET_CHR_2       Jump if "caps shift" on-
                                          ly pressed.
3C                   INC A
20F5                 JR NZ,GET_CHR        Jump unless no keys at
                                          all are pressed.
CD8E02  GET_CHR_2    CALL 028E,KEY_SCAN   DE: = keyboard scan.
7B                   LD A,E
FE27                 CP 27
28F8                 JR Z,GET_CHR_2       Jump if "caps shift"
                                          only pressed.
3C      INC A
28F5                 JR Z,GET_CHR_2       Jump if no keys at
                                          all pressed.
C9                   RET
```

Notice that there was not one but two loops in the above routine. The first loop waits *until* either "caps shift" only or no keys at all are pressed, and the second loop then waits for any key or key combination *other* than "caps shift only" or no keys at all. The purpose of having two loops instead of one is that the first loop waits until the human finger is removed from the previous key depression and the second loop waits for a new key. If this were not one you would have an awfully fast "repeat" facility (not desired).

In order to understand the subroutine below, I ought to explain what the data bytes in the message table were all about. 80 means "no input is required", 90 means "input Y or N" (for YES or NO) 92 means "input 0, 1 or 8", 95 means "input any digit or *enter*", 96 means "input any digit", 98 means "input any digit between 0 and 7", and 9E means "input either 0 or 1".

The subroutine below must be called in a rather special way — CALL MESSAGE/DEFB — where xx is a byte between 01 and 12. The choice of byte determines which message is printed. Here's the subroutine:

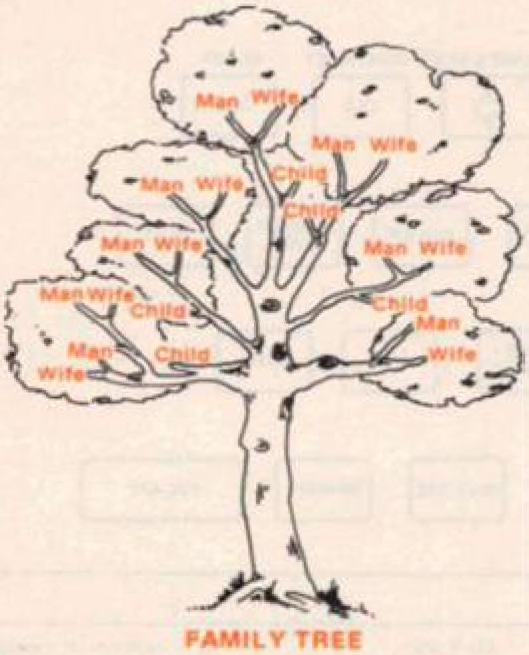
```
                     ORG DCCC
AF      MESSAGE      XOR A                   A: = 00
CD0116               CALL 1601,CHAN_OPEN     Use Stream number zero
                                             (lower part of screen).
E1                   POP HL                  HL: = address of data
                                             byte.
46                   LD B,(HL)               B: = number of message
                                             to print.
23                   INC HL                  HL: = subroutine return
                                             address.
E5                   PUSH HL                 Re-stack return address.
219FDB               LD HL,MESSAGES-1
7E      ME_SEARCH    LD A,(HL)
23                   INC HL
FE80                 CP 80
38FA                 JR C,ME_SEARCH          Find start of next
                                             message.
10F8                 DJNZ ME_SEARCH          Search to find appropriate
                                             message.
7E      ME_PRINT     LD A,(HL)               A: = next character of
                                             message.
23                   INC HL
FE80                 CP 80
C8                   RET Z                   Return if data byte 80
                                             found.
3003                 JR NC,ME_INPUT          Jump if other data byte
                                             found.
D7                   RST 10                  Print character.
18F6                 JR ME_PRINT
6F      ME_INPUT     LD L,A
26DB                 LD H,INP_TABLE high     HL: points to allowed
                                             inputs.
3E3F                 LD A,"?"
D7                   RST 10                  Print a question mark.
E5      ME_I_LOOP    PUSH HL
CDB5DC               CALL DCB5,GET_CHR       Wait for key to be pressed.
CD1E03               CALL 031E,KEY_TEST      Convert to character code.
E1                   POP HL
FE20                 CP20
2811                 JR Z, ME_EXIT           Jump if "ESCAPE"
                                             pressed. (Note that the
                                             stack is deliberately
                                             unbalanced).
E5                   PUSH HL
4F                   LD C,A                  C: = character just input.
7E      ME_CHECK     LD A,(HL)               A: = one of the allowed
                                             characters.
23                   INC HL
47                   LD B,A
E67F                 AND 7F                  Disregard bit seven.
B9                   CP C
2807                 JR Z,ME_EXIT            Jump if allowed character
                                             was input.
CB10                 RL B
30F4                 JR NC,ME_CHECK          Jump if there are more
                                             characters to check.
E1                   POP HL
18E3                 JR ME_I_LOOP            Try again.
E1      ME_EXIT      POP HL                  Balance the stack.
C36E0D               JP 0D6E,CLS_LOWER       Clear lower part of screen
                                             and return.
```

You can test this program out by writing the machine code CALL MESSAGE/DEFB ??/RET to any address and then calling it. If you put a PAUSE 0 instruction after the USR instruction then you will see the message at the bottom of the screen for as long as you need. Notice that "Light Screen Designer", "Number of Radians? " and "Machine Code Address?" will stay on the screen during PAUSE 0, but all the others will have been erased by the machine code as soon as the correct input was made.

In the next installment of this program I shall actually start doing something useful in the way of drawing pictures. Till then, SAVE all the material we've covered so far and we'll add to it with each new installment.

Toni Baker

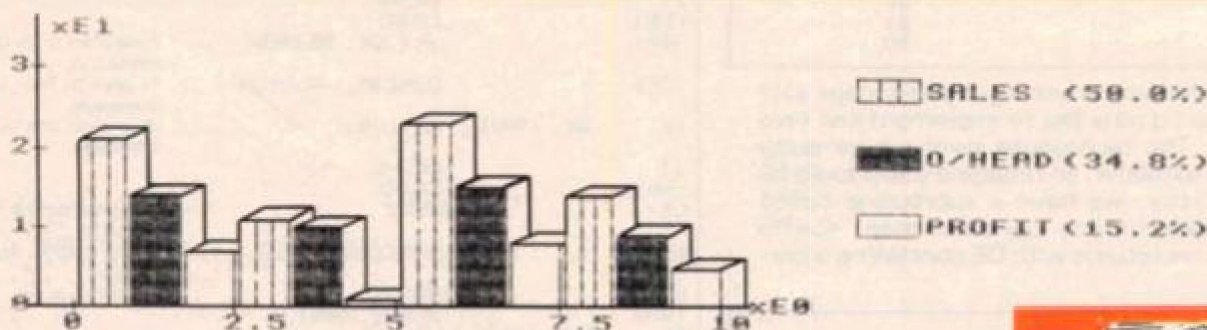
BEL GEN Advanced Genealogy Package

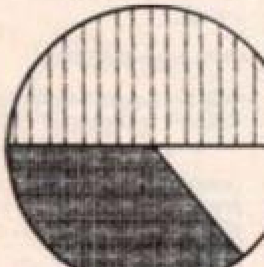




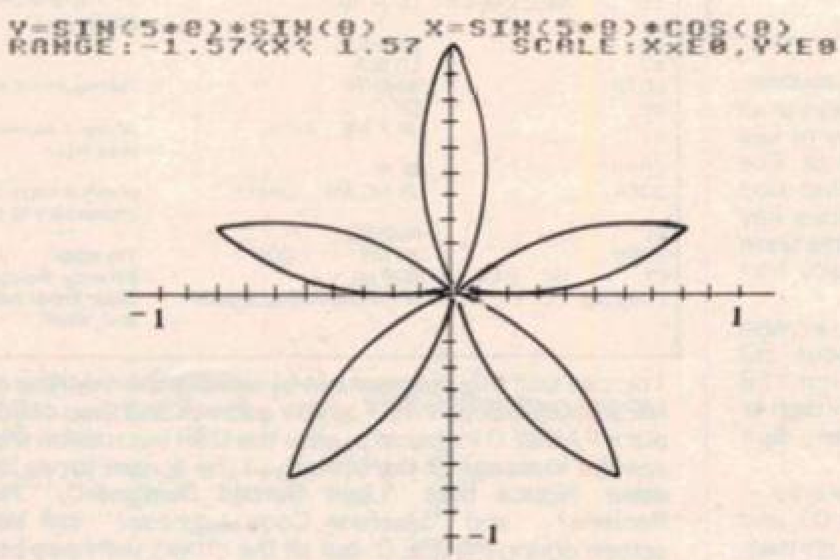
NEW

BEL GRAPH Advanced Graphics/Plotting





1. Keyboard/File Entry
2. Edit
3. Rename Records Default Settings
4. Line Graphs
5. Bar Charts/Histograms
6. Pie Charts
7. Vert Bar Charts
8. Data Entry and Plot
9. Function (Equation) Entry and Plot
10. Parametrics



Also:– **BEL BASE - DATA BASE**
CHEM ONE – 'O' LEVEL CALCULATIONS
CHEM TWO – 'O' LEVEL GENERAL CHEMISTRY

BEL TECH Limited, Stanmore Industrial Estate, Bridgnorth, Shropshire. WV15 5HP
Telephone: 07462 5420 (24 Hrs.)

BEL TECH



BEL TECH Software

Order Form	TAPE	QTY	DISC	QTY	TOTAL
BEL BASE	13.65		15.25		
BEL GRAPH	14.89		16.45		
BEL CHEM Chemistry 1	8.50		10.35		
Chemistry 2	8.50		10.35		
BEL GEN Geneology	12.50		13.99		

TOTAL inc. P. & P. £

NAME __________

ADDRESS __________

COMPUTER TYPE ______ MODEL ____

DISC (if applic.) ______

CREDIT CARD No. PAYMENT BY CHEQUE, P.O., VISA, ACCESS

Please Complete & Return to:-
BEL TECH Limited, Stanmore Industrial Estate, Bridgnorth, Shropshire. WV15 5HP, or
Telephone: 07462 5420 (24 Hrs.)

A New Dimension in Spectrum Hardware

Joystick and Interface

for Sinclair Spectrum with these features to give you endless hours of enjoyment.

1. Super positive response fire button.
2. Firm suction cups for stable one hand operation.
3. Snug fit hand moulded grip.
4. Additional fire button.
5. Extra long 4 ft lead.

The interface supplied with the Quick Shot (TM) has a two joysticks facility.

The first port simulates 6789 & 0 keys. The second port simulates in (31) command. It will run any Software.

1. Using keys 6, 7, 8, 9 and 0.
2. Having redifinable key function.
3. Using in (31) i.e. Kempston.
4. Any Software you write yourself.



£22.95

Light pen

The LIGHT PEN enables you to produce high resolution drawings on your own TV screen simply by plugging into the ear socket of your Spectrum. The controlling software supplied with the light pen has 16 pre-defined instructions. You can change colour (Border, Paper, Ink), draw circles, arcs, boxes, lines and insert text onto the screen at any chosen place, you can also draw freehand. There is a feature to retain the screens and animate. On the 48K Spectrum you can retain 5 screens. You can also use the machine code on its own in your own programs, for selecting out of a menu etc. The LIGHT PEN is supplied with a control interface, to adjust the sensitivity/pen alignment.

£19.95



Keyboard for use with a Spectrum or ZX81

£45.00

Our cased keyboard has 52 keys, 12 of which form a numeric pad. The 12 keys comprise 1-9 numeric plus full stop and shift keys, all in red, to distinguish from the main keyboard keys which are in grey, the keys contrast with the black case to form a very attractive unit. The case has been designed to take a ZX81 or Spectrum computer. 16K, 32K or 64K can also be fitted to the motherboard inside the case (81 model only). The case is also large enough for other add-ons like the power supply to be fitted, giving a very smart self-contained unit with which other add-ons e.g. printer etc. can still be used. Our ZX Professional keyboard offers more keys and features than any other model in its price range making it the best value keyboard available.



Spectra-Sound

The so-called speaker in your Spectrum is really on a 'buzzer'. With the DK Tronics "SPECTRA SOUND" you can generate fully amplified sound through the speaker on your TV set. SPECTRA SOUND is a very simple but highly effective add-on. This means that you no longer have a faint beep but a highly amplified sound, which can be adjusted with the TV volume control. The SPECTRA SOUND fits compactly and neatly inside the Spectrum case and is connected by three small crocodile clips.

£9.95

Please send me .... @ £.................

Please send me .... @ £.................

Please add on £1.25 for post and packing.

I enclose cheque/PO payable to DK Tronics total £..................

or debit my Access/Barclaycard No.

Signature ....

Name ....

Address ....

Send to: DK Tronics Ltd., Unit 6, Shire Hill Ind. Est., Saffron Walden, Essex CB11 3AQ. Tel: (0799) 26350 (24 hrs) 5 lines

dk'tronics

DK Tronics Ltd., Unit 6, Shire Hill Industrial Estate, Saffron Walden, Essex CB11 3AQ. Telephone: (0799) 26350 (24 hrs) 5 lines

Seriously now. . .

More useful devices for the Spectrum and ZX81.

When I completed last months project I thought perhaps a few lines to tie up the loose ends may be needed in this issue; as it happens, a whole new article is required.

For those deprived of the joy of our last issue, may I just mention that the Editor's Project stemmed from a desire to improve the print quality of our listings. From these humble beginnings a whole avalanche of peripherals descended upon my unwary head.

Being thus deluged, it soon become obvious that I needed somewhere to keep all these items. Having purchased my own portable TV and the Shinwa CP80 printer purely for producing the magazine (Taxman please note!) the pressure from the rest of the family to let them eat on the table now and then was becoming irritating.

Cabinet member?

After reading many publications, I ordered a computer cabinet from Marcol Cabinets. This arrived four days late but an apology was sent in good time and I had no hard feelings — I'd waited longer for my computer with no apology at all! Marcol have since assured me that the delay problem has been solved. Marcol make four varieties of cabinet priced from £39.99 for a basic open unit to £185.00 for a traditional English style (old fashioned) cabinet with real wood veneer finish. For reasons of price and family harmony I opted for the £79.95 model.

It arrived as a "flat pack" which meant that I had to put it together myself. For someone who equates 'Do it Yourself' with 'First Aid', the future did not bode at all well.

It pleases me to say that after spreading all the pieces out and enlisting the help of the entire Elder household, construction was pretty straightforwrd and, despite some of the bits being incorrectly labelled, its simple system of screws and blocks meant that it was soon installed.

Constructed of veneered chip, the cabinet is sturdy and looks a lot better than my previous system of spaghetti leads. There is plenty of room for most of my bits and pieces, tapes and books and when I've finished the doors keep it all out of sight. I particularly like the design of the sliding shelf which is just right for sitting comfortably when working. It is a bit on the large size, — 32.5" high, 36" wide, 16.75" deep but holds all my bits. The picture will give you a better idea of its looks than my descriptive powers.

A friend Dave, a DIY fanatic, produced a cabinet for his computer at a cost of about £17 but for anyone not into DIY I'd recommend this as a unit for your consideration.

Spectrum Interfaces

1 Morex Peripherals supplied me with their unit for testing complete with detached connector cable. This unit is different from the others as it has an RS232 interface as well as the Centronics that I needed.

There are two versions of their driver software V1 and V2; V1 is their standard program and V2 is the new improved all singing all dancing. . . Well anyway, it's a lot more user friendly and performs many of the setting up functions of an option program. Although still under test and not yet released, I was unable to cause it to crash or malfunction.

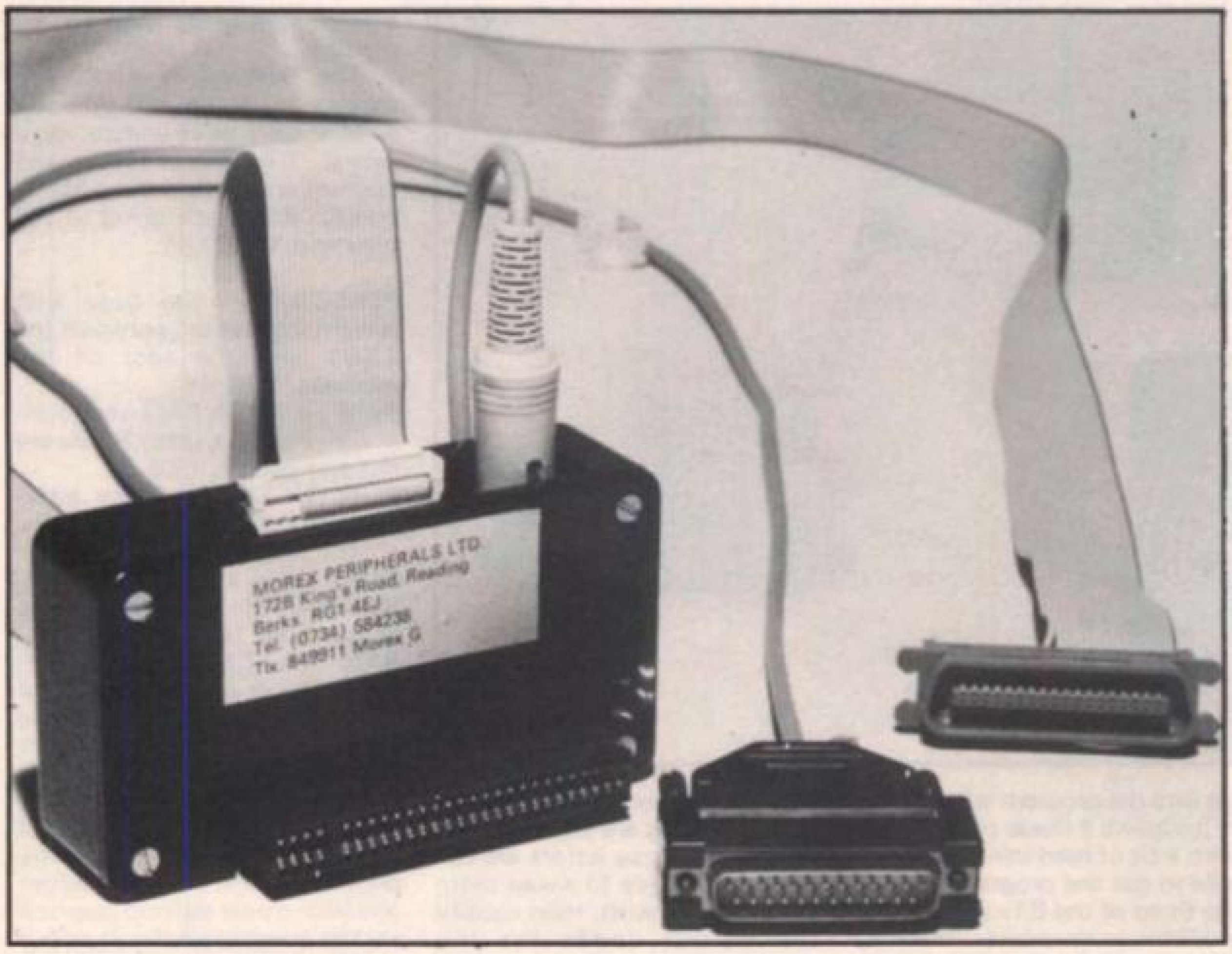
The instruction booklet/leaflets supplied were extensive and very detailed, but written for the experienced user rather than the beginner. The information includes all relevant details including pin connections and the interface is compatible with 'professional' programs such as Tasword II, Masterfile etc. As well as having the RS232, this was the only interface to feature a through port so that other units could be added on afterwards.

The Morex was the largest of the units that I tried, the case was roughly cut to accommodate the RS232 socket and the insides fitted loosely. Fitting the ribbon connector caused me a few headaches as I was loathe to use as much force to push it in as it eventually needed.

Now came the test. Software V2 loaded and ran without any problem and the manual gave full instructions for altering the codes which would be sent for UDGs and the Sinclair block graphic set. I had to write a short program to set up thee codes to suit my printer, I'm sure this could have been included in the V2 set up program, the default value of all these is 32 — a space.

Working with Tasword II there was no problem, producing screen dumps was easy both





in normal and double size and 80 column listings were simply a case of entering LLIST. Using the 32 column option on LLIST caused a slight problem as certain Keywords seem to cause a line of 31 characters, but this is only really a problem if you are trying to produce listings for a magazine. . .

I admit I know nothing about RS232 except that some printers and peripherals use it so I took it to the local expert. "Oh, a Morex" he said, "I use one to connect to my Beeb". The remainder of the evening was incomprehensible and I wished I'd brought an interpreter along. All I can report is that it worked and he was enthusiastic due to "full handshaking" capability — sounds like a Masonic ritual. If you require more details I suggest you contact Morex Perhipherals Ltd at 172B Kings Road, Reading, Berks RG1 4EJ (Tel: 0734 584238)

2 The ZXLprint III dropped through the door just in time for me to give it a good try out before writing this article and I'm very glad it did!

Supplied with a built in ROM instead of the usual software tape, the unit is small and the only one to lie flat, the others stand up on end. The cable is separate and connects to a set of pins exposed at the back. Next to these pins are a set of RS232 pins, but I really didn't get a chance to get expert help on this unit.

The instructions are brief but concise and simple, the printer can be controlled from one of the built in routines (most are catered for even the CP80!) or from external software. The latter is useful if you want to send special printer codes. It works with all the usual business software mentioned elsewhere. All the usual functions work perfectly, to get 32 characters per line a simple POKE will set it up. So why am I so impressed?

Well, options exist within the ROM to allow free use of LPRINT and LLIST as do the others, but COPY also is recognised and a screen dump is produced easily — all the others demand you call a machine code routine via USR.

This is useful but not exceptional, the real power lies in the fact that when it is put in graphics print mode, any time you LPRINT or LLIST a line with block graphics or UDGs the interface recognises them, shifts the printer into dot print mode, builds up the character on the paper in correct size relation (at least on the CP80) then changes back into text mode and continues with the line. This means that I can now reproduce ALL the Spectrum character set on listings.

There's only one thing that makes it less than perfect, but I must point out that ALL the interfaces have the same fault (or at least the ones I've tried) and that is that after the keyword THEN an extra space is printed, throwing the line one character out of sync with the screen.

I know that many readers like their programs to be printed in 32 character form so they can compare the screen with the listing; as you can see we have used this form and we will continue to do so. I'm sure that the intrusion of the extra space will not cause too much trouble. I will keep looking for the perfect interface but in the meantime I'll use the next best thing — the ZX Lprint III available from Euroelectronics, 26 Clarence Square, Cheltenham, Glos. GL50 2UJ.

Sometimes life gets tedious!

Having just written the above, yet another interface arrived. This one was the latest from Kempston and guess what? It performs exactly the same as the ZX Lprint III. Access to the Eprom is slightly different; the ZX Lprint is via a set of CHR$ codes, the Kempston Centronic E uses COPY:REM?

The Kempston stands upright and is in their usual sturdy case with a lip at the top. Both will produce screen dumps in normal or double size, but the Kempston builds it up sideways on the paper. Quite honestly, there is nothing between these impressive units in the operation of a Centronics printer, but the ZX Lprint has the added benefit of the RS232 connector.

Kempston Micro Electronics Ltd can be contacted at Unit 30, Singer Way, Woburn Road Industrial Estate, Kempston, Bedford MK42 7AF.

Softly softly. . .

Mentioned briefly last issue but now well tested is a print utility from Microdot Software of 30 Hazelmere Court, 26 Palace Road, Streatham Hill, London SW2 3NH. This was written and designed to be user friendly, and especially to be relocatable. Often a routine set at a particular area of memory gets in the way of other code but written in two parts, a text print and a screen copy routine, both or either of which can be relocated, this program solves the problem.

Primarily intended for the Cobra RS232 interface, it will work on most RS232 or Centronics interfaces with most of the popular printers. Written as compactly as possible to take up as little of the RAM as necessary, it has many features: unfortunately, sending user modified graphics and UDG codes is not included. Although not suitable for my applications, this is a neat and useful program, especially if you write a lot of long programs or need a lot of RAM for wordprocessing etc.

The instructions are very lengthy and very technical, an expert would find in it a wealth of detail, I found it confusing. Nevertheless, I was able to use it with only the minimum of head scratchig due to the on screen prompts. Even so, Kempston and ZX Lprint have rather eliminated the need for this program.

And finally

Lurking in a dusty corner of the office was a set of boxes, further investigation revealed a full Basicare system and a unit marked PERICON C Centronics interface.

Clutching my prize I rushed off and connected it to the ZX81 only to find there was no cable or instructions! A few calls to Basicare Microsytem Ltd at 5 Dryden Court, London SE11 4NH and a very helpful Mr Grimsby soon replaced the missing items.

All the ZX81 listings in this issue are produced by the Memotech interface which is completely hardware controlled but has so far functioned adequately. The Basicare unit is software controlled so I was excited by the thought that I could modify the program to send ▶



specific graphic printer codes. This system is predominantly an enthusiast's product and so a mass of technical information is supplied. Suffice it to say that although it is complex, it is written in such a way that I was able to make sense of it.

Sure enough, I was able to get into the program with the aid of printouts it made of itself and with a bit of hard thinking, I was able to get the program to print the three of the 81s graphic set available on my printer. The way it is set up is awkward for making listings but sensible if you are printing a lot of text — surely a ZX81 wordprocessor is pushing it a bit! Letters are printed lower case and inverse letters are upper case. I have to swap them over to start with, then modify the output codes for the graphics and set the line length to 32 by DIMming B$(33). I couldn't get it to print a pound sign though (code 129 on my printer).

The Basicare is a real enthusiast's set up and you will need at least three units to run a printer. I list them below and cannot recommend them highly enough if you are going all the way with the ZX81.

PERSONA — the base unit which 'interfaces' between the ZX81 and the rest of the modules. £30.25
RAM 16 — 16K RAM usable only with Persona, other RAMs are not compatible. £26.75
PERICON C Centronics Interface described above £41.75

Other units include RAM 08, MINIMAP (to extend address space from 64K to 1M byte) RAM 64, DROM, TOOLKIT, PERICON A & B (input/output devices). It is interesting to note that neither ZX81 units suffer from the intrusive 'space after THEN' problem.

So that is the present state of affairs, I'm limited by my printers graphic set. If I could get one with a user defined graphics set the quest would be at an end and listings would be perfect!

All donations gratefully received. . .

Competition

Unscramble the 'jumbogram' and win yourself some software!



The problem

As I was compiling a list of software houses on my trusty Spectrum, there was the ominous flicker of lights that indicates a power fluctuation and usually the end of my program. However, this time the result was unexpected. The computer continued to function but when I recalled the list of names, the result was as you can see in Fig. 1.

Inspiration struck and I thought "what a good idea for a competition!" The task, therefore, is to find all the names of the software houses in the jumble of words. Each letter must only be used *once* and all the software companies mentioned sell software for the ZX81, Spectrum or both.

To enter, unscramble the jumbogram and write the names on the form below. Fill in the coupon with your name and address and send it, in a sealed envelope, to our Golden Square address marked 'Names Competition'. Please do not forget to write on the back of the envelope the number of companies that you have deciphered, without this your entry will not be considered valid. Best of luck to you all and don't forget to indicate which machine you have when you enter!

The prizes

To please both Spectrum and ZX81 owners, the prizes this issue consist of forty sets of software, twenty for each machine, from Durell Software and Protek Computing Ltd.

Durell Software have kindly supplied us with twenty sets of each of their best selling programs, Harrier Attack, Jungle Trouble and the widely acclaimed, Scuba Dive. All these programs have been or are in the best seller charts and are definitely worth adding to your collection.

For ZX81 owners, the prizes have been donated by Protek, a company that has acquired a good name for itself by selling quality software for both the ZX81 and the Spectrum as well as a joystick interface. To the winners Protek are offering twenty sets of their superb ZX81 arcade games, UFO and Byter.

The rules

- This competition is open to all UK and Northern Ireland readers of *ZX Computing*, except employees of Argus Specialist Publications Ltd, their printers and distributors, employees of Durell Software, Protek Computing Ltd, or anyone associated with the competition.
- As long as the correct coupon is used for each entry, there is no limit to the number of entries from each individual.
- All entries must be postmarked before July 31st, 1984. The prizes will be awarded to the first forty entries picked at random which have the correct answers, the decision to be made by the Editor of *ZX Computing*. No correspondence will be entered into with regard to the results and it is a condition of entry that the Editor's decision is accepted as final.
- The winners will be *notified* by post and the results will be published in a future issue of *ZX Computing*.

Results

Thank you for all the entries we received to the Electronic Stars Competition in the Feb/Mar issue of *ZX Computing*. Unfortunately, there is only one prizewinner, but *someone* has to win!

Congratulations to Ms Suzi Yann of Dudley in the West Midlands who is the lucky winner of the Alphacom printer kindly donated by Dean Electronics. Thank you again for all the entries and better luck next time!




```
QOENHAENEOCPEAMEMMTTSPSOUILTCRPA
EALSTUGAGELEOROKIDEIDLISUESITCIA
ENFOERQSTPMMHLUCNOEVRNLSVCICREIR
TIBHDVNECEFALHTIILIIKDOIITHREUDM
GTVAARIFOAISMSDULNAIMATAICARLPRO
SEFTPRKEMEGALINGT
```

Address your entries to:

ZX Computing Names Competition
1 Golden Square London W1R 3AB

I have found the following software companies in the jumbogram: ....

....

....

....

....

....

....

Name ....

Address ....

....

.... machine ....

Mozart c1984!

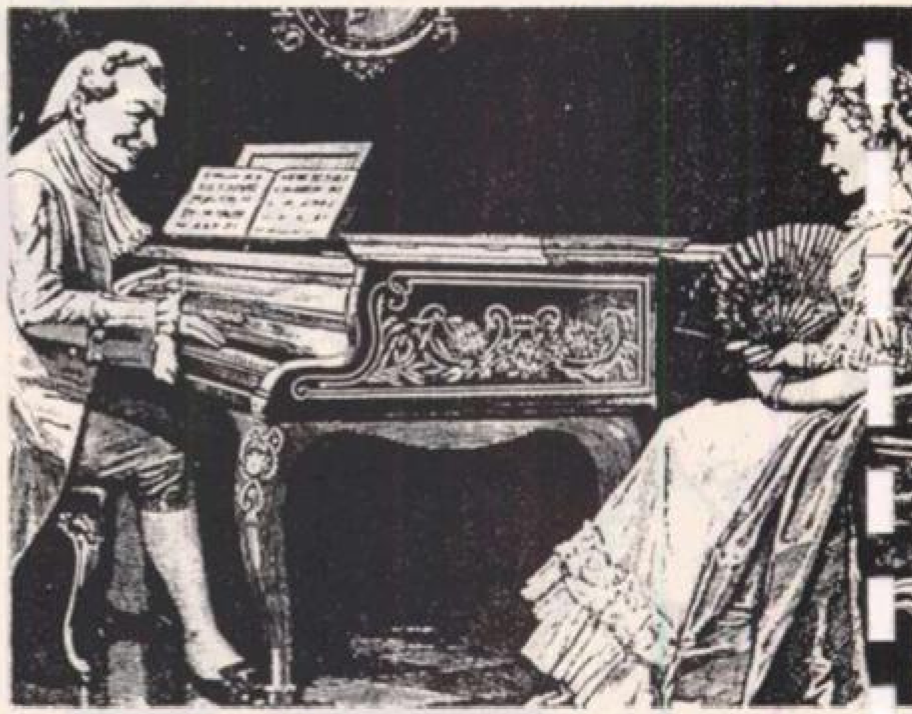
Chi-Yeung Choy of Ruislip flexes his fingers and performs this musical masterpiece.

The Spectrum's oft maligned beep has been considered too feeble for any reasonable musical application in the past, but now we present a program which will make you think again!

Instead of the usual selection of sound effects or brief burst of barely recognisable tunes, Chi-Yeung has successfully programmed a computer version of the first movement of Mozart's piano sonata in C Major K545. What is even more amazing is that it is written all in BASIC and fits into the 16K machine!

A great deal of thought and attention to technical detail has gone into this program for instance, the subroutines have been put at the start of the program to get every ounce of speed from the computer.

Chi-Yeung tells us that his greatest problem was fitting it into the 16K due to the amount of DATA required and that he almost gave up. Type it in and then give your fingers a rest and your ears a treat. Play on maestro!



Program breakdown

Lines 30-230	Subroutines.
Lines 240-520	DATA statements containing pitch of notes.
Line 1000	Set up variables for note duration.
Lines 1010-2070	Main program consisting of FOR-NEXT loops.

```
Ø>REM *  C-Y Choy   1983  *
  1Ø GO TO 5ØØ
  2Ø REM **********************
  3Ø REM SUBROUTINES
  4Ø REM **********************
  5Ø READ B,C,D
  6Ø BEEP 2*X,B: BEEP X,C: BEEP
X,D: RETURN
  7Ø FOR M=1 TO 16
  8Ø READ B: BEEP Z,B
  9Ø NEXT M: RETURN
 1ØØ FOR N=1 TO 8
 11Ø READ B: BEEP Y,B
 12Ø NEXT N
 13Ø BEEP X,14: BEEP X,19: BEEP
X,7: PAUSE 25
 14Ø RETURN
 15Ø READ B,C,D,E,F
 16Ø BEEP 2*X,B: BEEP Z,C: BEEP
Y+Z,D: BEEP Z,E: BEEP Y+Z,F: RET
URN
 17Ø FOR N=1 TO 12
 18Ø READ B: BEEP Z,B
 19Ø NEXT N: RETURN
 2ØØ BEEP X,19
 21Ø FOR N=1 TO 12
 22Ø READ B: BEEP Z,B-12
 23Ø NEXT N: RETURN
 24Ø READ B,C,D
 25Ø BEEP 2*X,B+5: BEEP X,C+5: B
EEP X,D+5: RETURN
 5ØØ REM **********************
 51Ø REM MAIN PROGRAM
 52Ø REM **********************
 53Ø LET X=1/2: LET Y=X/2: LET Z
=X/4: LET A=X/8
 54Ø GO SUB 5Ø
 55Ø READ B,C,D,E
 56Ø BEEP X+Y,B: BEEP Z,C: BEEP
Z,D: BEEP X,E: PAUSE 25
 57Ø GO SUB 5Ø
 58Ø BEEP X,19
 59Ø FOR N=1 TO 3
 6ØØ BEEP A,19: BEEP A,17: NEXT
N
 61Ø BEEP A,16: BEEP A,17: BEEP
X,16: PAUSE 25
 62Ø FOR N=1 TO 5
 63Ø READ B: BEEP Y,B
 64Ø FOR M=1 TO 14
 65Ø READ B: BEEP Z,B: NEXT M: N
EXT N
 66Ø GO SUB 7Ø
 67Ø GO SUB 1ØØ
 68Ø FOR N=1 TO 2
 69Ø FOR M=1 TO 4
 7ØØ IF N=1 THEN  BEEP Z,1: BEEP
Z,2: GO TO 72Ø
 71Ø BEEP Z,Ø: BEEP Z,2
 72Ø NEXT M: NEXT N
 73Ø RESTORE 212Ø
```

```
740 FOR N=1 TO 2
750 RESTORE 2120
760 READ B,C,D,E,F,G,H,I,J,K,L,
M,P
770 BEEP Y,B: BEEP Y,C: BEEP X+
Y,D: BEEP Z,E: BEEP Z,F: BEEP Y,
G: BEEP Y,H: BEEP A,I: BEEP A,J:
 BEEP A,K: BEEP A+Z,L: BEEP Z,M:
 BEEP X,P: PAUSE 50
780 NEXT N
790 FOR N=1 TO 4
800 GO SUB 70: NEXT N
810 GO SUB 150
820 BEEP Z,20: BEEP X+Z,21: BEE
P A,23: BEEP A,21: BEEP A,20: BE
EP A,21: BEEP Y,24: BEEP Y,21: B
EEP Y,24: BEEP Y,21
830 BEEP Y,23: BEEP Y,19: BEEP
2*X,26: BEEP Z,24: BEEP Z,23: BE
EP Z,21: BEEP Z,19
840 FOR N=1 TO 15
850 BEEP A,23: BEEP A,21: NEXT
N
860 BEEP Z,19: BEEP Z,21: BEEP
X,19
870 GO SUB 170
880 RESTORE 2180: GO SUB 200
890 BEEP X,7: BEEP X,23: BEEP X
,19: PAUSE 25
900 BEEP X,7: GO SUB 170
910 RESTORE 2190: GO SUB 200
920 FOR N=1 TO 2
930 GO SUB 70: NEXT N
940 BEEP X,17
950 RESTORE 2190
960 FOR N=1 TO 12
970 READ B: BEEP Z,B-5: NEXT N
980 BEEP X,14
990 RESTORE 2190
1000 FOR N=1 TO 12
1010 READ B: BEEP Z,B-17: NEXT N
1020 RESTORE 2220
1030 FOR N=1 TO 7
1040 GO SUB 70: NEXT N
1050 RESTORE 2030: GO SUB 240
1060 READ B,C,D,E
1070 BEEP X+Y,B+5: BEEP Z,C+5: B
EEP Z,D+5: BEEP X,E+5: PAUSE 25
1080 GO SUB 240
1090 BEEP X,24
1100 FOR N=1 TO 3
1110 BEEP A,24: BEEP A,22: NEXT
N
1120 BEEP A,21: BEEP A,22: BEEP
X,21: PAUSE 25
1130 FOR N=1 TO 4
1140 READ B: BEEP Y,B+5
1150 FOR M=1 TO 14
1160 READ B: BEEP Z,B+5
1170 NEXT M: NEXT N
1180 BEEP 2*X,21: PAUSE 25: BEEP
 X,21
1190 BEEP 2*X,19: PAUSE 25: BEEP
 X,19
1200 BEEP 2*X,17: PAUSE 25: BEEP
 X,17
1210 BEEP 2*X,16: PAUSE 25: BEEP
 X,16
1220 RESTORE 2300
1230 GO SUB 70
1240 RESTORE 2100
1250 GO SUB 70: GO SUB 100
1260 FOR N=1 TO 2
1270 FOR M=1 TO 4
1280 IF N=2 THEN  BEEP Z,5: BEEP
 Z,7: GO TO 1300
1290 BEEP Z,6: BEEP Z,7
1300 NEXT M: NEXT N
1310 FOR N=1 TO 2
1320 RESTORE 2120
1330 READ B,C,D,E,F,G,H,I,J,K,L,
M,P
1340 BEEP Y,B-7: BEEP Y,C-7: BEE
P X+Y,D-7: BEEP Z,E-7: BEEP Z,F-
7: BEEP Y,G-7: BEEP Y,H-7: BEEP
A,I-7: BEEP A,J-7: BEEP A,K-7: B
EEP A+Z,L-7: BEEP Z,M-7: BEEP X,
P-7: PAUSE 50
1350 NEXT N
1360 FOR N=1 TO 2
1370 FOR M=1 TO 16
1380 READ B: BEEP Z,B-7
1390 NEXT M: NEXT N
1400 FOR N=1 TO 2
1410 FOR M=1 TO 16
1420 READ B: BEEP Z,B+5
1430 NEXT M: NEXT N
1440 RESTORE 2290
1450 GO SUB 150
1460 BEEP 2*X,21: BEEP Z,20: BEE
P Y+Z,21: BEEP Z,20: BEEP Y+Z,21
1470 BEEP Y,19
1480 RESTORE 2310
1490 FOR N=1 TO 14
1500 READ B: BEEP Z,B: NEXT N
1510 FOR N=1 TO 15
1520 BEEP A,16: BEEP A,14: NEXT
N
1530 BEEP Z,12: BEEP Z,14: BEEP
X,12
1540 RESTORE 2180
1550 FOR N=1 TO 12
1560 READ B: BEEP Z,B-7: NEXT N
1570 BEEP X,12
1580 RESTORE 2180
1590 FOR N=1 TO 12
1600 READ B: BEEP Z,B-19: NEXT N
1610 BEEP X,0: BEEP X,16: BEEP X
```

```
,12: PAUSE 25
2ØØØ REM **********************
2Ø1Ø REM DATA
2Ø2Ø REM **********************
2Ø3Ø DATA 12,16,19,11,12,14,12
2Ø4Ø DATA 21,19,24
2Ø5Ø DATA 9,11,12,14,16,17,19,21
,19,17,16,14,12,11,9
2Ø6Ø DATA 7,9,11,12,14,16,17,19,
17,16,14,12,11,9,7
2Ø7Ø DATA 5,7,9,11,12,14,16,17,1
6,14,12,11,9,7,5
2Ø8Ø DATA 4,5,7,9,11,12,14,16,14
,12,11,9,7,5,4
2Ø9Ø DATA 2,4,5,7,9,11,13,14,9,1
1,13,14,16,17,19
21ØØ DATA 21,23,24,23,21,19,17,1
6,17,19,21,19,17,16,14,12
211Ø DATA 11,19,16,12,14,19,16,1
2
212Ø DATA 26,23,19,21,23,21,19,2
1,19,21,19,18,18
213Ø DATA 26,-1,2,7,11,26,23,19,
16,Ø,4,7,12,16,19,16
214Ø DATA 24,-3,Ø,6,9,24,21,18,1
4,-1,2,6,11,14,18,14
215Ø DATA 23,-5,-1,4,7,23,19,16,
12,-3,Ø,4,9,12,16,12
216Ø DATA 21,-6,-3,2,6,21,18,14,
11,-5,-1,2,7,19,14,11
217Ø DATA 9,11,12,15,16
218Ø DATA 19,14,19,23,26,23,19,2
3,24,21,18,21
219Ø DATA 19,14,19,22,26,22,19,2
2,24,21,18,21
22ØØ DATA 7,-17,-15,-14,-12,-1Ø,
-8,-6,-5,19,22,21,19,17,16,14
221Ø DATA 13,-15,-13,-11,-1Ø,-8,
-6,-4,-3,25,28,26,25,22,21,19
222Ø DATA -7,2,4,5,7,9,11,13,14,
2,5,4,2,Ø,-1,-3
223Ø DATA -4,11,12,14,16,18,2Ø,2
1,23,-4,-1,-3,-4,-7,-8,-1Ø
224Ø DATA -12,21,28,26,24,23,21,
19,17,2,9,7,5,4,2,Ø
225Ø DATA -1,19,26,24,23,21,19,1
7,16,Ø,7,5,4,2,Ø,-1
226Ø DATA -3,17,24,23,21,19,17,1
6,14,-1,5,4,2,Ø,-1,-3
227Ø DATA -4,16,23,21,2Ø,17,16,1
4,12,-3,Ø,-1,-3,-5,-7,-8
228Ø DATA -1Ø,1Ø,14,12,1Ø,9,7,5,
4,5,7,9,1Ø,12,14,16
229Ø DATA 14,13,14,13,14
23ØØ DATA 14,2,4,5,7,9,11,13,14,
9,11,13,14,16,17,19
231Ø DATA 21,23,24,26,28,26,24,2
3,21,19,17,16,14,12
```

news

In .... Brief

• Timedata Ltd of 16 Hemmells, High Road, Laindon, Basildon, Essex SS15 6ED have produced two programs, both of which are utilities. TT-S consists of five programs in one packet to provide a versatile toolkit and costs £7.95,HI-T is a set of routines for formatting the screen. Up to 32 lines of 64 characters can be displayed, and input positioning to any part of the screen, window definition, pixel printing and definable comma spacing for flexible tabulation are its functions. A useful program which costs £5.95.

• Bel Tech have produced a series of programs for the serious user; Bel Base, a comprehensive Data Base program, BelGraph, a versatile graph plotting program, Bel Chem 1 & 2, an 'O' level revision program and Bel Gen, for keeping Genealogical records (if you happen to be tracing your family tree!) More details (including the price) can be obtained from Bel Tech Ltd at Stanmore Industrial Estate, Bridgnorth, Shropshire WV15 5HP.

• Fowler Software has marketed two tapes for graphics programming; Graphics Subroutine and Draw 15. Draw 15 seems to be for static art and the other is for animation. They also sell Giant's Dinner, a neat multi-choice childrens' program which ably demonstrates the capabilities of the other two programs. Fowler Software can be contacted at Hendon Mill, Nelson, Lancashire BB9 8AD.

• Procom Software's Dynamic Graphics, a two cassette package, is also aimed at the screen artist. I had a quick look at this one and it appears to be a very powerful tool; 27 functions for character designing and animation by using Sprites (fast machine code automatic moving objects) and 50+ for creating Hi Res screen drawings.

The manual was complex and needs careful reading, but the whole unit is designed as a professional programmers aid and as any good craftsman will tell you it takes time to learn how to use the tools of the trade.

• This editions most unusual program has got to be 40 Sermon Outlines for Busy Preachers from Melvyn Hamilton at 4 Southlands Avenue, Louth, Lincs. LN11 8EW for a mere £5.00. No it's not a joke (I promise) but all the computer fanatic priests who spend their time programming instead of organising their sermons will really find this one useful. The gospel according to St Sinclair?

• Widgit Software of48 Durham Road, East Finchley, London N2 9DT have produced an interesting tape containing two programs. Humpty Dumpty Mystery is on side one and consists of two games of differing difficulty levels and Who Killed Cock Robin is on the reverse side. (I'm sure I've heard those names somewhere before!) The program requires you to deduce the culprit in both cases by a combination of sensible questions and intelligent guesses and is a great program to encourage logical thinking. The age range is for those just starting to read up to ten and the price is £6.00.

• The Eastmead Medical Series bows in with The Complete Guide to Medicine, First Aid and How Long Have You Got. The first two cost £6.90 and the last, a real party killer, is £3.45. Both are available from Eastmead House, Lyon Way,Camberley, Surrey GU16 5EZ.

• Orwin Software, well known for their excellent ZX81 collections, venture into the Spectrum market with Education Compendium, a set of games and programs with an educational bias. An interesting series of Logo tortoise routines are on side two.The package cost £6.00 from 26 Brownlow Road, London NW10 9QU.

Automata UK OK?

The cult of Pimanism is growing rapidly, a subversive organisation guaranteed to strike terror into the hearts of the strongest microprocessor chip. We wait with baited breath for the next onslaught.

Meanwhile, since our last issue, three forays into the realms of arcade games have escaped from their offices. Pi Eyed is a sort of farcical frogger plus. The old Piman is under your control and has to drink as much as he can. As in real life, the more he drinks the more awkward his movements become. Interesting.

Olympimania could have been great with the sporting interest at the moment, five events are billed but four of these turn out to be the same game with different obstacles. Once again you control the Piman. Personally, I found it too fast at the start to let me get into the game easily and thereafter boring. Still, one man's meat...

Pi Balled is Automata's version of Q Bert and this time you control a character called Bert (surprise, surprise) who has to visit every square on the pyramid of Pi to change them to his colour while the Piman and associates hinder you. A simple but annoyingly addictive game, for me this is definitely the best of the batch.

All are for the 48K Spectrum and cost £6 each from Automata UK Ltd, 27 Highland Road, Portsmouth, Hants. PO4 9DA.

It's magic

What have Paul Daniels and Silversoft Ltd in common? The answer lies in Silversofts Linkword language courses. Long, long ago when I was involved in the entertainment industry, I was introduced to a 'Memory Man' routine.

This routine was performed by the MM being blindfolded and members of the audience shouting out words which were written on a board under numbers 1 to 20 (or 40 if you were really good). Once the twenty or forty words were written, members of the audience called out the numbers and the MM instantly told them the word written under it. The secret was in the preparation, the MM had to spend ages memorising object for each of the numbers: for instance, 15 could be related to a rabbit. When the word for each number was called out the MM had to make a quick link between the object related to the number and the word. For example, if the word for number 15 was HAT, the MM would be associating 15 with rabbit and so create a mental image of a rabbit wearing a hat! This link of a silly mental image works. This is where Paul Daniels comes in, he used the same routine on one of his recent shows.

This is the basis of these Linkword language courses in French, Spanish, German and Italian. OK so it doesn't teach the pronunciation or grammatical peculiarities of the language, but it will give you an easy and enjoyable way of learning a basic set of keywords which will enable you to communicate after a fashion on your holiday this year.

The courses consist of audio and computer cassette and instruction booklet and cost £12.95. Does it work? Ask Paul Daniels.

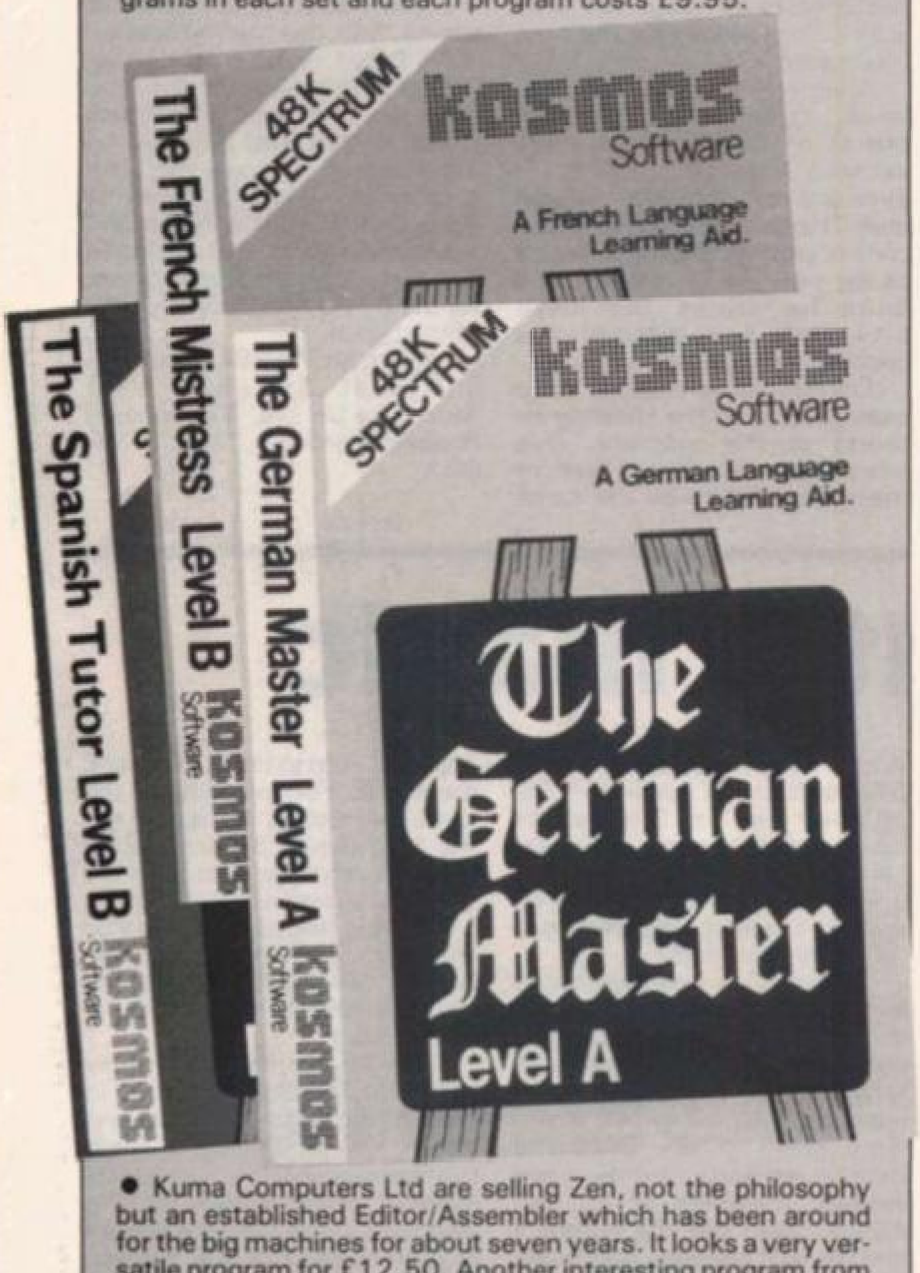
• Hickstead/Mathstead, Jungle Adventure and Diamond Quest are the titles of three Games for Girls, from CCS. They seem to me to have a distinct lack of respect for the female intelligence level. It's a good intention, but I'm doubtful of the method used.

Five other programs have also been released, all of which continue in CCS' business and strategy vein. CCS have been specialising in this type of game since the early days of the ZX81. The titles are War 70, Pacific War, Manage, Oligopoly and Stockmarket. All programs cost £6.00 with the exception of Diamond Quest which is a pound cheaper. CCS can be contacted at 14 Langton Way, Blackheath, London SE3 7TL.

• Grand old masters of the market, Artic, have released Monkey Bizness, another Kong game and Bear Bovver, a walls and ladders style game.

Both are up to their usual high standard with good graphics but Bear Bovver is my favourite with superb animation. For the 48K machine they are priced at £5.95 and £6.95 respectively.

• Kosmos Software of 1 Pilgrim Close, Dunstable, Bedfordshire LU5 6LX have given their French Mistress some company with the release of two new language cassettes, the German Master and The Spanish Tutor. There are two programs in each set and each program costs £9.95.



• Kuma Computers Ltd are selling Zen, not the philosophy but an established Editor/Assembler which has been around for the big machines for about seven years. It looks a very versatile program for £12.50. Another interesting program from them is Logo. If you can't wait for David Nowotnik's other articles or want a machine code version to really explore this language then this could be the one for you! It costs £9.95.

Finally, a game by the intrigueing name of Vegetable Crash which is apparently selling well in Japan. It sounds like a down to earth Galaxians, but I can't comment as they have not seen fit to send a review copy yet. At £6.95, they are available from 12 Horseshoe Park, Pangbourne, RG8 7JW.

Hurg to a classic adventure

Melbourne House have ventured from old to new with two recent releases.

Classic Adventure is the micro version of the original 1970s Fortran game and, as a text only standard format adventure game, is a departure from their advanced style graphics/text multi-instruction game The Hobbit!

Even more of a departure and a tremendous advance in games programming is HURG, or High-level, User-friendly, Real-time, Games-designer. (No wonder they shortened it to HURG!)

I must admit to being impressed, and with the volume of software that I check each week that takes some doing! This is a very powerful program and yet everything can be controlled by a joystick. To achieve this versatility, extensive use of menu-driven routines is employed. You move through a set of procedures from which you may diverge to access any feature you wish to use in your game. Animation can be achieved easily and to a very high standard, and with options to detect collisions and special events this is very flexible.

Demonstration programs to show 'Zap, Chase and Jump' style games are also given on the tape and if you can produce a good game why not enter the £3000 competition for the best game designed using this system, the closing date has been extended to June 30th so you've still got time to write a masterpiece!

If you have great ideas for games but are limited by a lack of programming skill, this could be the answer. However, it won't do all the work for you and, like any tool, you will have to learn to use it but Melbourne House have made learning as easy as possible.

HURG will set you back £14.95 and Classic Adventure a mere £6.95

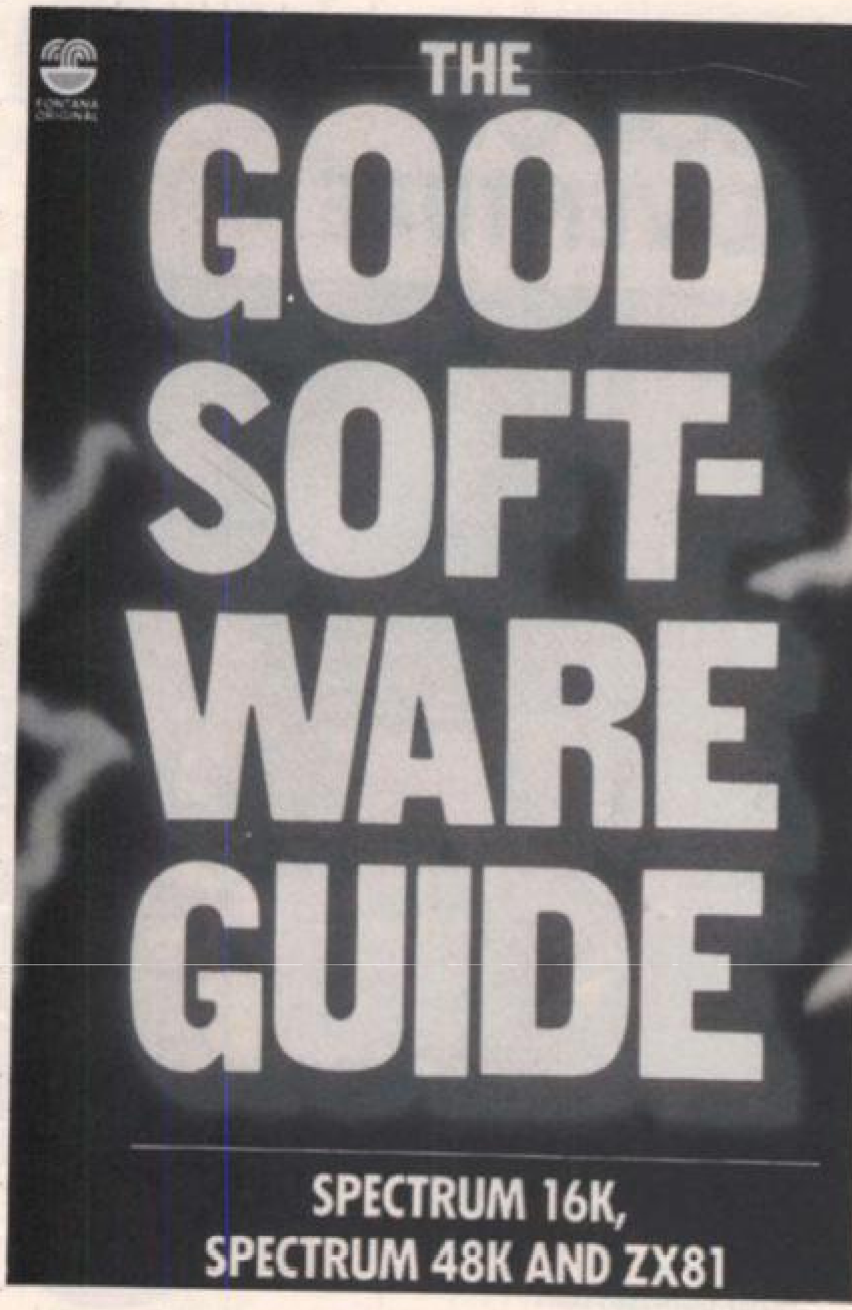
Support for the 81

A big welcome to Softchoice Ltd of 52 Platts Lane, London NW3 7NT from myself and probably many ZX81 users. Softchoice is a subsidiary of International Publishing & Software Inc, a large organisation who have been producing software for quite a while in Canada. I am informed that response to the adverts placed so far has been very good indeed so the '81 still lives! Before you all write in confirming this, regular readers will know that I am a supporter of the old ZX81 and need no encouragement to continue to back it.

The range of programs is wide and comprehensive, Signals & Code and Tiny Logo are two general interest programs and three games are offered, Mr. Munchee, Speed Snake and Wrath of Kong. These are clearly based on arcade games and I'm sure you need no further explanation. Fastload is a utility to enable you to save and load programs up to six times faster than normal and Fastword is a word processing package would you believe!

Finally, Diet is a domestic program to help you watch your eating habits. The Spectrum, however, is not entirely ignored and a version of Diet is available plus two educational games specifically for the 16/48K Spectrum, Take Away Zoo and Adding Machine, aimed at the 5-9 age range. Signals & Code, Mr. Munchee and Speed Snake cost £4.95 each, Tiny Logo, Wrath of Kong and Diet are £5.95 and Fastload and Fastword are priced at £7.50. All the Spectrum programs are £6.95 each.

Book of reviews



Alongside the boom in microcomputers, the software on sale has naturally proliferated. It is impossible for us to keep up with the amount released each issue, although we do try. Software checklists, although useful, do tend to be merely lists. Helpful though is our sister magazine, Software Today, dedicated entirely to reviews of software for all types of domestic micro.

Along these lines is the Good Software Guide from Fontana Paperbacks which is described as a "source of reliable expert, comprehensive and critical information". The intention is to include only the best of the available software, I would be interested to know what criteria they use when deciding on the 'best'. Nevertheless, over 300 programs are covered, divided into nine groups: adventure, arcade, collections, educational, novel, practical, strategy, traditional and utility and forty reviewers of a wide range of age and ability were used in the production of the book.

Fontana plan to make this an annual publication and there will be three editions, the first of which is devoted to the Sinclair machines — ZX81, 16 and 48K Spectrum. A good idea when the results of a recent survey indicates that Sinclair has 55% of the home market.

The 290 pages in the review issue still seemed too few compared to the amount on the market but for any keen games player, future historian or just for general interest it is well worth considering at £3.95 in the shops.

In . . . . Brief

- Pixel Pads could be very useful to Spectrum programmers who like to present their work with carefully drawn screens. Pixel Pads are large sheets of gridded paper in the Spectrum's screen format.

  Each pad contains 50 sheets on each of which is a large pixel grid, several 8 × 8 UDG grids and a summary of related information such as INK and PAPER colour numbers. They may be obtained from Peter Bamford & Co at 10 Barley Mow Passage, Chiswick, London W4 4PH for £4.95 per pad.

- If you are near Battersea Park on the 14th or 15th of July, why not look in to the What Micro? Dr Barnardos Show. As well as Micros there will be other events and various surprise celebrities from the world of entertainment. This could be one way of combining a family day out and your hobby!

- Scott Adams is to adventure what Tim Hartnell is to Microbooks. We look forward to receiving the first in a new range of programs for review from the British branch of his company, Adventure International.

  Marvel comic characters will be involved in what is described as graphic adventure games, the first of which is the Incredible Hulk. A comic will accompany the tape and at a certain point you take over . . .

  All we need now is for Walt Disney to get in on the act.

A mixed bag



CDS Micro Systems have recently released a selection of programs covering a wide range of topics. In the games field there is Winged Warlord, a jousting game, Magic Meanies, a chase game and Timebomb. All are 16/48K games and Timebomb has the added benefit of being compatible with the Currah Microspeech unit if you have it.

Spectrum Safari is for 48K machines only and is described as an adventure-style game. This looks suspiciously like the game of the same name marketed by A J Rushton last year, not that it's anything to object to, I liked that game. It's a collection of mini games tied together with the theme of escaping from an island.

And finally, for those going abroad this year there is French is Fun and German is Fun, both of which are designed for students or for brushing up on a language before venturing abroad. A light touch is used with pictures and words. I can't give a personal comment as CDS so far have not seen fit to supply us with review samples.

In . . . . Brief

• Euroelectronics, of 26 Clarence Square, Cheltenham, Glos. GL50 2JP, as well as producing the excellent ZX Lprint III detailed in my project, can supply software to print screen dumps in four colours on either the MCP40 or Tandy CGP printers.

It is called COPY 4, and a BASIC version is also supplied which can be adapted for use with earlier Lprint versions or other makes of interface.

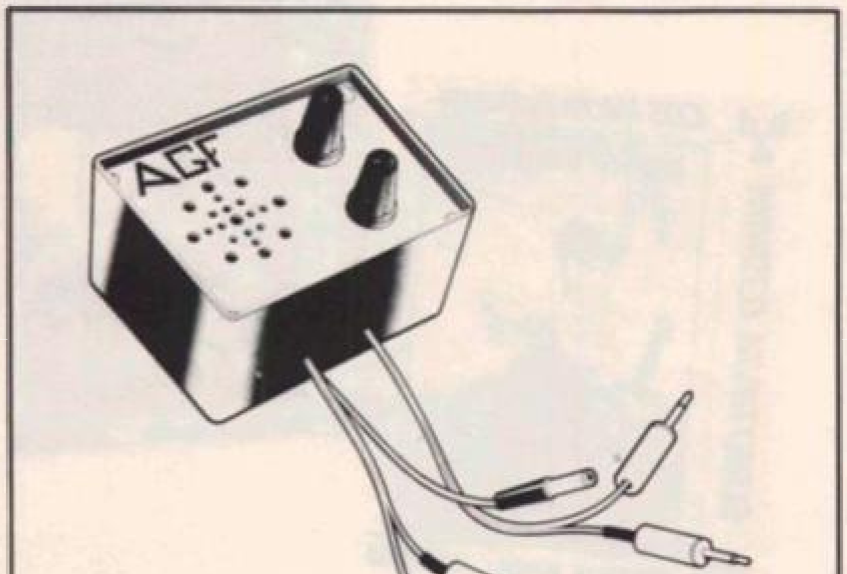
Priced at £5.50 it provides an economic way of getting colour into your print.

• An interface which is not only fully programmable but also includes a sound amplifier has been marketed by Rainbow electronics of Glebe House, SouthLeigh, Witney, Oxon OX8 6XJ. The cost is £24.00 plus £1 per p&p. We will tell you more in a future issue.

• The Ram Tubo is a combination interface which provides a joystick port and a ROM cartridge software port. The specifications look good, although it does not appear to be programmable and the provision of a through port to allow for other peripherals to be added on afterwards is a useful facilty.

The cost if £22.95 plus £1.00 p&p from RAM Electronics (Fleet) Ltd., 106 Fleet Road, Fleet, Hampshire.

Save! Save! Save!



AGF have reduced the price of their joystick interfaces to £26.95 for the programmable and £13.95 for module II (plus of course the obligatory £1.00 p&p). Both of these have undergone slight modification to allow the Quickshot II rapid "auto fire" feature to be implimented.

A nifty new item also marketed by AGF is Amplisound, a lead switching and amplifier unit. This cuts out the changing of leads when loading and saving. A volume control is included on the amp, much to the relief of many parents! The volume control also contains an on/off switch which operates the computer, this is very useful indeed. The price was not finalised at the time of going to press, but should be around £24.95 + £1.00 etc.

Adventurous reading



Melbourne House, famous for The Hobbit, have recently published. The Computer and Video Games Book of Adventure by Keith Campbell, resident expert from the magazine of the same name.

In this his first book, Keith introduces adventure games, gives advice on how to solve and write them and explains their history. The listings in the book are suitable for Spectrum, BBC and Commodore 64 machines and teaches simple programming techniques to control movement, objects, space and time and interpret inputs. The book includes a full listing for a complete adventure in all three formats plus an introduction from Scott Adams, generally regarded as the father of adventures on the Micro.

Most of the programming tips appear to be expanded versions of his earlier C&VG columns but are still valid and nigh on essential for the serious adventure programmer. Available from most bookshops at £5.95.

Dib dib?



One company that treats the Sinclair computers with some respect is Microvitec who have produced two colour monitors for use with the Spectrum and the new QL.

The Spectrum compatible monitor is claimed to be the only Low Complexity Colour Display equipped to handle Spectrum outputs. Code numbered the CUB 1431/MZ, it is set in a metal cabinet with a die cast frame surround finished in matt black or beige. Should you decide to upgrade to the QL or another computer with a RGB/TTL colour output then there is a socket on board to accept output in this form. The CUB 1431/MZ has a 14in screen and costs £225 (plus VAT and p&p).

Specifically for the QL (and other Hires 80 colour machines) is their latest colour monitor, the CUB 1451/MQ3. This features a high contrast, self-converging PIL tube (no, I don't know what it means, but it sounds good!) with a resolution of 653 × 585 addressable pixels.

Both units are supplied with a dedicated cable and the controls are sited at the rear of the cabinet. The CUB 1451/MQ3 will cost you a mere £299 (plus VAT and p&p).

In........................Brief

• The Arcade professional joystick, a big and sturdy unit, seems to have had a successful initial run. So much so in fact that Emax Computing have not only reduced the price to £28.95 but are giving a full refund to those who purchased it at the original price.

Just send your sale invoice to Emax Computing, Pinfold Lane Industrial Estate, Bridlington, N Humberside. I wish other companies would adopt this refund idea. . .

• A floppy disc system for the Spectrum is available from Technology Research Ltd of 356 Westmount Road, London SE9 1NW. The FDC-1 MK2 is the interface and will support two 40, 80 track, single or double side drives. The cost is £85.00 exc. VAT, + £1.00 for p&p. To compliment this they will provide you with a 40 track single side drive with PSU for £185.00 exc. VAT + £5.00 p&p.

• For the ZX81, Sircal Instruments (UK) Ltd of 11 Southfields Court, Sutton Common Road, Sutton, Surrey SM1 3HJ provide a low cost EPROM Programmer. Send an sae to the above address for more details if you are interested in this unit which works with Intel 2716, 2732 or similar EPROMs.

ZX Microdrives on the shelf!

By now, the well sought after ZX Microdrives should be available in your local Smiths, Boots, Menzies or other major retail outlets which has got to be good news for Spectrum owners!

The Microdrive stores a minimum of 85K bytes' information on magnetic cartridge but average access time to information is a mere 3.5 seconds as opposed to a frustrating 3.5 minute wait. It is controlled by the ZX Interface 1 which can support up to eight microdrives, giving a total of 680 bytes capacity. As the Interface 1 also incorporates an RS232 interface, your Spectrum can be linked to other computers and peripherals and a local area network facility which will link upto 64 Spectrums.

Both the ZX Microdrive and the ZX Interface 1 are priced at £49.95 each and cartridges are available separately at £4.95.



In .... Brief

● BASIC and FORTRAN in Parallel, written by S J Wainwright & A Grant is published by Bernard Babani and is another of their small but good value books.

The introduction claims that it could be used to learn FORTRAN or BASIC, or both languages at the same time! I had enough difficulty learning BASIC on its own. At £1.95 it looks a useful book which includes a 16K FORTRAN interpreter in BASIC.

● Tim Hartnell is probably one of the most prolific of the microcomputer authors and has recently released his latest offering via Interface Publications. Non-machine-specific, it is intended to take you through programming your computer step by step. The Easy Way to Program Your New Computer costs £3.95 for 208 pages including programs to try.

● Spectrum Graphics and Sound by Steve Money, published by Granada at £6.95 deals in depth with these Specific areas of the machine. Or rather, in depth with nine chapters on graphics and briefly with one chapter on sounds. Looks worth a glance if you are interested in this application.

● Also from Granada is Exploring FORTH by Owen Bishop. Even though the book is non-machine-specific, a great deal of examples and information has been packed in and the author has taken pains to ensure that most of them will work with whichever FORTH system you have. £6.95 from bookshops.

● Another language book from Granada is Introducing Logo by Boris Allen priced at £5.95. This book seems to deal specifically with using the language, the philosophy and Turtle graphics. LOGO is arousing a good deal of interest at the moment and this might be a timely release.

● For the younger would-be programmer, Steve Betts has written Spectrum Magic. Published by W. Foulsham & Co Ltd at £4.95, it is written in simple style it will be useful for those who really feel they want a 'Computing by Numbers' course.

The trouble is that the vocabulary needed is necessarily greater than the age range it is aimed at. If you can read and understand words such as Variables, Randomize, Fractional then what's the point in using topics like Dolls and Toys? Nevertheless, a colourful, concise introduction.

Give 'em some stick!





Kempston, probably as near to an industry standard as is likely in the field of joysticks/interfaces, have produced two new models of joystick. Modestly called the PRO 1000 and PRO 3000, the old but evergreen model being given pride of place as the PRO 5000, these will no doubt receive much interest from gamers everywhere.

The PRO 1000 has eight way leaf switches, a five foot cable so you won't have to sit right on top of your television, and a bar instead of a button which allows for left or right hand play — a good idea. This idea is used again on the PRO 3000, in fact the bases are identical but this one features an 'ergonomically designed' grip handle (honest, their wording-not mine) with both a top and front fire button so you can use either thumb or trigger finger to fire. Kempston are giving you no excuse for missing that vital universe-saving shot!

Priced at a reasonable £10.99 for the PRO 1000, £12.75 for the PRO 3000 and, of course, £13.80 for the old faithful PRO 5000, these should be available from your local zap shop.

Sinclair snooker



Everything goes to pot in this program from Tim Blewitt in the West Midlands.

Sinclair Snooker is a textual simulation in which the computer makes the decisions on the outcome of each turn. Just in case there are any readers who have never played the game or watched it on television here are the rules:

Each player must take turns to try and pot a red ball and, if successful, then pot a coloured ball (except red!). The balls are worth the following points:

RED	1
YELLOW	2
GREEN	3
BROWN	4
BLUE	5
PINK	6
BLACK	7

If you make a foul shot — miss or hit the wrong colour ball — then you lose a minimum of four points. A snooker is when the next player cannot get a clear shot at the correct colour ball and you can attempt a snooker yourself at any time.

On the ball

When all 15 balls have been potted, the computer will give you instructions on how to pot the remaining colours in their correct order. You will be given the angle between the white, the colour and the pocket, and you must input the speed (or strength) of the shot — this may include decimals.

The success of this shot is determined by the computer by matching your input with two numbers. If it is between these numbers — known only to the computer — then you succeed. The higher the angle, the harder it is to pot due to the numbers being closer together — a small random factor has been introduced here to add to the difficulty. This routine can be modified by looking at the variables T and X.

There are three skill levels and this affects your opponent's chances of potting a colour after potting a red. The ratios of success to level is as follows:

Level 1	1 in 3
Level 2	2 in 3
Level 3	always

Variables

Here follows a key to the variables used in Sinclair Snooker.

P	Player's score.
C	Computer's score.
B	Current break (player or ZX).
R	Number of reds left.
SK	Skill level.
V	Difficulty level for potting colours.
M	Decides which colour the computer pots.
K	Success of snooker attempt.
G	Decides angle for shot at colour.
E	Player's speed (strength) of shot.
T/X	Upper and lower limits for speed input.
N	Value of colour.
FL	How many points awarded for a foul shot.
TB	Decides who wins the tie break.

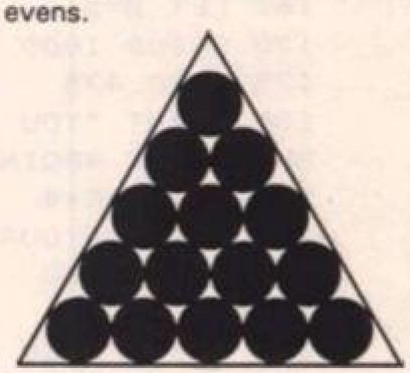
Other variables are used for loops etc.

Structure

Here follows a brief description of the program:

0-269	Player attempting a red ball.
273-474	Player attempting a coloured ball.
475-600	Player attempting a red ball.
1000-1185	ZX attempting a red ball.
1190-1300	ZX attempting a coloured ball.
1500-1570	Player's snooker attempt.
2000-2342	Player's attempt at final colours.
2343-2420	Winner routine.
2500-2620	ZX's attempt at final colours.
3000-3030	Print out scores.
3500-3630	ZX attempting a coloured ball.
4000-4140	Player's snooker attempt at final colours.
4500-4540	Instructions for potting final colours.
4700-4760	Tie break winner.
5000-5100	Title.

Your chances of potting a colour are determined by the value of the ball, it will always be easier to pot yellow than black, but of course this means fewer points. If the game ends in a draw then a tie break is called and the odds are set at evens.



```
   5 GOSUB 5000
  10 LET R=0
  15 LET B=0
  20 LET P=0
  30 LET C=0
  31 PRINT "SKILL LEVEL?(1-3)"
  32 INPUT SK
  40 PRINT "DO YOU WANT TO BREAK
? (Y/N)"
  50 INPUT A$
  60 IF A$="N" THEN GOTO 82
  70 IF A$="Y" THEN GOTO 90
  80 GOTO 40
  87 GOSUB 1000
  88 GOTO 475
  90 LET D1=INT (RND*6)
 100 LET D2=INT (RND*6)
 110 LET D3=INT (RND*6)
 115 IF R=15 THEN GOTO 2000
 120 IF D1=3 AND D2=3 OR D1=3 AN
D D3=3 OR D2=3 AND D3=3 THEN GOT
O 180
 130 IF D1<=1 OR D2<=1 OR D3<=1
THEN GOTO 240
```

```
140 PRINT "YOU DID NOT POT A RE
D."
150 PRINT "YOUR BREAK WAS: ";B
154 PAUSE 300
155 CLS
157 GOSUB 3000
160 LET B=0
170 GOSUB 1000
175 GOTO 475
180 PRINT "YOU COMMITTED A FOUL
SHOT-LOSE 4POINTS."
190 LET C=C+4
200 PRINT "YOUR BREAK WAS: ";B
204 PAUSE 300
205 CLS
207 GOSUB 3000
210 LET B=0
220 GOSUB 1000
230 GOTO 475
240 LET R=R+1
250 PRINT "YOU POTTED A RED."
260 LET B=B+1
262 LET P=P+1
265 PRINT "YOUR BREAK IS NOW: "
;B
267 PAUSE 300
268 CLS
269 GOSUB 3000
273 PRINT "DO YOU WANT TO TRY F
OR A COLOUR OR ATTEMPT A SNOOKER
?"
274 PRINT "TYPE:""C""FOR COLOUR
"
275 PRINT "     ""S""FOR SNOOKE
R"
276 INPUT Q$
277 IF Q$="C" THEN GOTO 280
278 IF Q$="S" THEN GOTO 1500
279 GOTO 272
280 PRINT "WHICH COLOUR DO YOU
WANT TO TRY FOR?"
290 INPUT S$
295 LET N=INT (RND*7)
300 IF S$="YELLOW" THEN GOTO 36
1
310 IF S$="GREEN" THEN GOTO 363
320 IF S$="BROWN" THEN GOTO 365
330 IF S$="BLUE" THEN GOTO 367
340 IF S$="PINK" THEN GOTO 369
350 IF S$="BLACK" THEN GOTO 371
360 GOTO 280
361 LET V=2
362 GOTO 380
363 LET V=3
364 GOTO 380
365 LET V=4
366 GOTO 380
367 LET V=5
368 GOTO 380
369 LET V=6
370 GOTO 380
371 LET V=7
380 IF N+V<=7 THEN GOTO 440
390 PRINT "YOU DID NOT POT THE
";S$
400 PRINT "YOUR BREAK WAS: ";B
404 PAUSE 300
405 CLS
407 GOSUB 3000
410 LET B=0
415 IF R=15 THEN GOTO 2500
420 GOSUB 1000
430 GOTO 475
440 PRINT "YOU POTTED THE ";S$
450 LET B=B+V
460 LET P=P+V
470 PRINT "YOUR BREAK IS NOW: "
;B
472 PAUSE 300
473 CLS
474 GOSUB 3000
475 IF R=15 THEN GOTO 2000
540 PRINT "DO YOU WANT TO TRY F
OR A RED OR ATTEMPT A SNOOKER?"
550 PRINT "TYPE:""R""FOR RED"
560 PRINT "     ""S""FOR SNOOKE
R"
570 INPUT W$
580 IF W$="R" THEN GOTO 90
590 IF W$="S" THEN GOTO 1500
600 GOTO 475
1000 LET D1=INT (RND*6)
1010 LET D2=INT (RND*6)
1020 LET D3=INT (RND*6)
1030 IF D1=3 AND D2=3 OR D1=3 AN
D D3=3 OR D2=3 AND D3=3 THEN GOT
O 1090
1040 IF D1<=1 OR D2<=1 OR D3<=1
THEN GOTO 1140
1050 PRINT "I DID NOT POT A RED.
"
1060 PRINT "MY BREAK WAS: ";B
1064 PAUSE 300
1065 CLS
1067 GOSUB 3000
1070 LET B=0
1080 RETURN
1090 PRINT "I COMMITTED A FOUL S
HOT-YOU GAIN4 POINTS."
1100 PRINT "MY BREAK WAS: ";B
1104 PAUSE 300
1105 CLS
1107 GOSUB 3000
1110 LET B=0
1120 LET P=P+4
1130 RETURN
1140 LET R=R+1
1150 PRINT "I POTTED A RED."
```

```
1154 LET B=B+1
1155 LET C=C+1
1160 PRINT "MY BREAK IS NOW: ";B
1170 PAUSE 300
1180 CLS
1185 GOSUB 3000
1190 LET M=INT (RND*6)+2
1195 GOSUB 3500
1200 IF M=2 THEN PRINT "I POTTED
 THE YELLOW."
1210 IF M=3 THEN PRINT "I POTTED
 THE GREEN."
1220 IF M=4 THEN PRINT "I POTTED
 THE BROWN."
1230 IF M=5 THEN PRINT "I POTTED
 THE BLUE."
1240 IF M=6 THEN PRINT "I POTTED
 THE PINK."
1250 IF M=7 THEN PRINT "I POTTED
 THE BLACK."
1260 LET B=B+M
1270 LET C=C+M
1280 PRINT "MY BREAK IS NOW: ";B
1284 PAUSE 300
1285 CLS
1287 GOSUB 3000
1290 IF R=15 THEN GOTO 2500
1300 GOTO 1000
1500 LET K=INT (RND*2)
1510 IF K=0 THEN GOTO 1550
1520 PRINT "YOUR SNOOKER ATTEMPT
 WAS       UNSUCCESSFUL."
1522 LET B=0
1524 PAUSE 300
1525 CLS
1527 GOSUB 3000
1528 IF R=15 THEN GOTO 2500
1530 GOSUB 1000
1540 GOTO 475
1550 PRINT "YOU FORCED ME TO MAK
E A FOUL      SHOT-YOU GAIN 4 POIN
TS."
1552 LET B=0
1554 PAUSE 300
1555 CLS
1560 LET P=P+4
1565 GOSUB 3000
1567 IF R=15 THEN GOTO 2000
1570 GOTO 475
2000 DIM J$(7,6)
2005 GOSUB 4500
2010 LET J$(2)="YELLOW"
2020 LET J$(3)="GREEN"
2030 LET J$(4)="BROWN"
2040 LET J$(5)="BLUE"
2050 LET J$(6)="PINK"
2060 LET J$(7)="BLACK"
2070 LET N=2
2080 PRINT "DO YOU WANT TO TRY F
OR THE      ";J$(N);" OR ATTEMPT
 A SNOOKER?"
2090 PRINT "TYPE:""C""FOR COLOUR
"
2100 PRINT "      ""S""FOR SNOOKE
R"
2110 INPUT Z$
2120 IF Z$="C" THEN GOTO 2145
2130 IF Z$="S" THEN GOTO 4000
2140 GOTO 2080
2145 CLS
2146 GOSUB 3000
2150 PRINT "YOU ARE NOW TRYING T
O POT THE    ";J$(N)
2155 LET G=INT (RND*90)
2157 PRINT
2160 PRINT "THE ANGLE IS:";G;" D
EGREES."
2170 PRINT "INPUT YOUR SPEED NOW
(1-6)"
2180 INPUT E
2190 IF E<1 THEN GOTO 2170
2200 IF E>6 THEN GOTO 2170
2210 LET T=INT (G/15+RND)
2220 IF E>T THEN GOTO 2250
2230 PRINT "YOUR SHOT WAS TOO SO
FT AND YOU  MISSED THE ";J$(N)
2235 PRINT "YOUR BREAK WAS: ";B
2236 PAUSE 300
2237 LET B=0
2238 CLS
2239 GOSUB 3000
2240 GOTO 2515
2250 LET X=INT (RND*5+2)
2260 IF E>X THEN GOTO 2300
2270 PRINT "YOU POTTED THE ";J$(
N)
2275 LET P=P+N
2277 LET B=B+N
2278 PRINT "YOUR BREAK IS NOW: "
;B
2280 LET N=N+1
2285 IF N>7 THEN GOTO 2340
2290 GOTO 2080
2300 PRINT "YOU MISSED THE ";J$(
N)
2310 PRINT "YOUR BREAK WAS: ";B
2320 LET B=0
2322 PAUSE 300
2324 CLS
2325 GOSUB 3000
2330 GOTO 2515
2340 CLS
2342 GOSUB 3000
2343 IF P>C THEN GOTO 2370
2345 IF P=C THEN GOTO 2363
2350 PRINT "BAD LUCK,I WON THIS
FRAME."
2360 GOTO 2380
```

```
2363 GOSUB 4700
2365 GOTO 2380
2370 PRINT "WELL DONE,YOU WON TH
E FRAME."
2380 PRINT "DO YOU WANT ANOTHER
GAME?(Y/N)"
2390 INPUT Y$
2400 IF Y$="Y" THEN GOTO 10
2410 IF Y$="N" THEN STOP
2420 GOTO 2380
2500 DIM J$(7,6)
2501 LET J$(2)="YELLOW"
2502 LET J$(3)="GREEN"
2503 LET J$(4)="BROWN"
2504 LET J$(5)="BLUE"
2505 LET J$(6)="PINK"
2506 LET J$(7)="BLACK"
2507 GOSUB 4500
2508 LET N=2
2515 LET L=INT (RND*2)
2517 IF L=1 THEN GOTO 2560
2520 PRINT "I DID NOT POT THE ";
J$(N)
2530 PRINT "MY BREAK WAS: ";B
2540 LET B=0
2550 GOTO 2080
2560 LET B=B+N
2570 LET C=C+N
2580 PRINT "I POTTED THE ";J$(N)
2590 PRINT "MY BREAK IS NOW: ";B
2592 PAUSE 300
2594 CLS
2596 GOSUB 3000
2600 LET N=N+1
2610 IF N>7 THEN GOTO 2340
2620 GOTO 2515
3000 PRINT AT 0,0;"CHALK-YOUR-CU
E    SHARP-SHOOTER"
3005 PRINT AT 1,2;"CHALLENGER
 V    SINCLAIR"
3010 PRINT AT 2,6;P
3020 PRINT AT 2,25;C
3023 PRINT
3024 PRINT "REDS LEFT=";15-R
3025 PRINT
3030 RETURN
3500 LET H=INT (RND*6)
3505 IF SK=1 THEN GOTO 3530
3510 IF SK=2 THEN GOTO 3570
3520 RETURN
3530 IF H<=3 THEN GOTO 3600
3540 RETURN
3570 IF H<=1 THEN GOTO 3600
3580 RETURN
3600 PRINT "I MISSED A COLOUR."
3610 PRINT "MY BREAK WAS: ";B
3620 LET B=0
3630 GOTO 475
4000 LET SN=INT (RND*2)
4002 IF N<=4 THEN GOTO 4006
4004 LET FL=N
4005 GOTO 4010
4006 LET FL=4
4010 IF SN=0 THEN GOTO 4080
4020 PRINT "YOUR SNOOKER ATTEMPT
 WAS        UNSUCCESSFUL."
4030 LET B=0
4040 PAUSE 300
4050 CLS
4060 GOSUB 3000
4070 GOTO 2515
4080 PRINT "YOU FORCED ME TO MAK
E A FOUL     SHOT-YOU GAIN ";FL;"
 POINTS."
4085 IF N=7 THEN GOTO 2340
4090 LET B=0
4100 PAUSE 300
4110 CLS
4120 LET P=P+FL
4130 GOSUB 3000
4140 GOTO 2080
4500 PRINT "ALL OF THE REDS HAVE
 BEEN          POTTED. THE COLOURS
MUST NOW BE POTTED IN ORDER."
4501 PAUSE 500
4502 CLS
4503 PRINT "TO POT A COLOUR YOU
MUST JUDGE  THE SPEED WITH WHICH
 YOU HIT THESHOT."
4505 PRINT "         I WILL GIVE
YOU THE      ANGLE BETWEEN THE PO
CKET, THE    COLOURED BALL AND TH
E CUE-BALL. THE HIGHER THE ANGLE
,THE MORE    SPEED YOU WILL NEED
TO POT THE   BALL.HOWEVER,THE MOR
E SPEED YOU USE,THE LESS YOUR CH
ANCE OF      POTTING THE BALL WIL
L BE."
4510 PAUSE 1000
4520 CLS
4530 GOSUB 3000
4540 RETURN
4700 PRINT "THE FRAME ENDED IN A
 DRAW";
4710 LET TB=INT (RND*2)
4720 IF TB=0 THEN GOTO 4750
4730 PRINT " BUT   YOU LOST THE
TIE-BREAK."
4740 RETURN
4750 PRINT " AND   YOU WON THE T
IE-BREAK."
4760 RETURN
5000 PRINT AT 0,8;"sinclair snoo
ker"
5010 PRINT AT 1,8;"BY TIM BLEWIT
T"
5020 PRINT
5100 RETURN
```









Ref. CC.100 — Especially designed for your home computer system whilst also giving plenty of storage space for software, magazines or perhaps a video recorder. Keeps your equipment **dust free.**

Standard Features!

1. Smooth sliding shelf locks in position just when you require it — and neatly glides away after use.
2. The computer shelf has been designed to allow adequate 'Knee room' for you to sit comfortably, and will avoid 'programmers back-ache'.
3. Adequate room to position tape recorder, printer, disc-drives, etc. alongside computer for ease of use.
4. All cables neatly and safely out of sight, while allowing all units to be permanently connected if desired.
5. Lots of storage space for cassettes, cartridges, books, listing paper, joysticks, etc.
6. With shelves and doors closed, instantly becomes an elegant piece of furniture.
7. The lower shelf suitable for your video recorder or storage of software, etc.
8. Smart Teak effect finish.
9. Supplied as a flat-pack. Very simple to assemble, using just a screwdriver. Full instructions supplied.
10. Measurements: Height 32½ in., Width 36 ins., Depth 16¾ ins.

PRICE only £79.95 incl. VAT.

Ref. DLC.200 — This cabinet has the same basic features as model CC.100 but comes in a real wood veneer finish. Built in traditional English style to a very high standard of workmanship this cabinet will grace any home. Available in teak, oak, mahogany or walnut with brass fittings.

PRICE £185 incl. VAT — fully assembled.

Ref. BHU 300 — A basic home computer housing unit in a teak effect finish. Storage shelf for magazines etc. Supplied as a flat-pack, very simple to assemble. Full instructions supplied. Measurements — 32'' wide x 31'' high x 18'' depth plus VDU bridging unit 21'' x 6'' x 12'' (available as a separate unit). See **Ref. BU.500** below.

PRICE only £46.95 incl. VAT

Ref. BHU 400 – Similar to BHU 300 but without the storage shelf. Measurements — 32'' wide x 27'' high x 18'' depth plus VDU bridging unit 21'' x 6'' x 12''

PRICE only £39.99 incl. VAT.

Ref. BU.500 — Bridging Unit only — Price £12.50 incl. VAT

TERMS OF OFFER:– UK Mainland Customers only. Please allow 28 days for receipt. Cash with order or charge to one of the credit card accounts specified. Money back if not satisfied provided goods are returned undamaged at the customers expense to Marcol Cabinets, Southampton within 72 hours of taking delivery.

ORDER FORM

Please supply me with the following cabinets:–

Ref. No.	Qty.	Colour	Price	Total
....	......	....	£......... each	£ ....
....	......	....	£......... each	£ ....
			Delivery Charge	£ 5.00
			Grand Total	£

I enclose my cheque for £ .... or please debit my

Access/Barclaycard No. ....

Name ....

Signature ....

Address ....

....

....

MARCOL CABINETS, PO BOX 69, SOUTHAMPTON SO9 7EQ. TEL: (0703) 731168 (24hr ANSWERING)

The ZX81 soft selection

Nick Pearce gives us his critical views on some of the latest ZX81 software.

Bubble Bugs and Bank Robber — Romik Software

I thoroughly enjoyed Bubble Bugs. It is, I suppose, an invaders-type game. Your base ship is at the bottom left hand corner of the screen (where it remains during the game) and is under attack from the bubble bugs. You race backwards and forwards along the bottom of the screen firing missiles and "popping" the bubbles. The bubble bugs drop bombs which can damage both your craft and your base ship. Your ship can tolerate a certain amount of damage but eventually is destroyed whereupon the game ends.

Not only do bubble bugs bomb you but when hit, they shatter and the debris can also inflict damage. When your craft is hit, a rescue vehicle is automatically sent out from the base ship to carry out repairs. You consume a unit of fuel each time you fire a missile and as you can only carry ten units, you have to return to the base ship regularly to replenish stocks. Alternatively, the rescue craft will come to your aid should you run out completely.

You can select an attack speed from 1 (easy) to 10 (impossible) the screen display is good and the action is smooth and responsive. Quite an impressive game.

In some respects Bank Robber is similar. You rush along the bottom of the screen taking money from the bank on the left and depositing it in your house on the right. As usual, life just isn't that easy. You operate in a street constantly bombarded by meteorites which you must avoid, and occasionally bricks and pills drop from the sky. If you catch a brick, a protective building will be rebuilt and a pill gives you another 'life'.



The protective buildings resemble the shields in an invaders type game. It is worth trying to keep them built (the meteorites quickly destory them) in order to facilitate your dashes between the bank and your house. Unfortunately, you can only enter your house or the bank when the doors are open: they occasionally close during the game.

There are twenty levels of play and the density of the meteorites is selectable from 1 to 3. Another good quality action game from Romik Software.

Both Bank Robber and Bubble Bugs are 16K games. They include on screen instructions and each has a "freeze frame" whereby if the action all gets too much you can freeze everything while composing yourself.

Asteroids! — Mikro-Gen

This is a fast moving arcade-type action game. The screen displays large variously shaped asteroids moving through space and endangering your ship. By rotating your ship and moving forwards, you try to avoid them, or shoot them with your lasers. When hit, an asteroid breaks into several smaller ones. There are also UFOs and space mines to contend with.

You have three lives per game. Game speed is selectable in ten steps from slow to impossibly fast. Your score and the highest score recorded, are displayed at the end of each game.

ZX Asteroids is good fun to play although the graphics are not particularly subtle. The action is generally good and the game is responsive at high speed. Control of your space craft takes a while to master, it keeps moving forwards indefinitely once the thrust key is pressed and the only way to stop is to rotate through 180 and thrust forward again.

Not possessing a joystick, I had to use the ZX81 keyboard although a joystick option is included. It is a 16K game and is recorded both sides of the cassette.

Forty-niner and Asteroids — Software Farm

If you are still prevaricating over whether to purchase the ZX Spectrum, maybe you should forget it and get Forty-niner instead. Forty-niner is an excellent game. It requires a little practice to master, but persevere, it's worth it. No sound or colour but the high resolution screen display gives another dimension to the trusty ZX81.

The 1949 Great American Gold Rush has just started and you are out there with the best of them excavating for the precious metal. Somewhat unusual hazards (for a gold digger anyway) are present to hinder you. Giant rats search through your excavations after you and there are snakes which when released, head straight for the surface and destroy everything in their path. There is also a gremlin to contend with and you have to deposit ex-

cavated soil on the surface to delay it capturing you.

The aim of the game is to collect all the nuggets of gold. When you have accomplished this awesome task, you pass on to a more difficult stage with more rats but less snakes to destroy them.

An interesting feature of Forty-niner is the facility for the player to select his own control keys. A hi-score chart is kept by the computer and displayed at the end of each game and there are five levels of play. The hi-res display is excellent, those rats really do look like rats. A great game.

The Software Farm game of Asteroids is similar to the Mikro-Gen version reviewed above. It has three levels of play, and you have three lives per game. It requires a 16K RAM pack and the graphics are in "normal" ZX81 resolution. There is an option for up to four players to hold a contest, each player's highest score being separately recorded and identified in a score table.

I thought the action rather jerky, even at the hardest level but the game works well and is good fun nonetheless. More expensive than Mikro-Gen's game but with the additional facility to keep score during a contest.

Gamestape 1 — Fawkes Computing

There are four good quality games on this cassette; Dodge-it and Trogan Dragon on the A side and Death Trap and Tablets of Hippocrates on the reverse.

Dodge-it is a variation of the hungry blob-type game. Your character moves anticlockwise following lanes around the screen. Your only control is to change lanes when you reach gaps by the use of the cursor keys. True to tradition you consume dots and leave crumbs. The second time you travel the same lane you eat crumbs and leave dots and so on. There is also a strawberry which, if eaten, gives you extra points and, of course, the inevitable monster which travels clockwise around the lanes and must be avoided.

This is an unusual but effective game in which your character is always on the move. You cannot reverse direction to escape the monster — once you are in the same lane that's it. Good fun to play but frustratingly difficult to judge which lane to cross to at the next gap. Dodge-it has four speed levels from learner to expert and a high score feature.

In Trojan Dragon the programmer has taken some liberties with popular legend to produce an interesting game.

The problem is that you have to guard your castle at two positions simultaneously. You control the drawbridge but must only allow "goodies" across and the swordsmen on the battlements must only kill "baddies". The difficulty is in recognisinig goodies and baddies as they approach these positions so that appropriate action can be taken.

An interesting idea is in the way the game is made more difficult for the experienced player; at the harder levels of play the goodies and baddies are represented by characters which more closely resemble each other. There is also a choice of speed but the speed increases automatically as the game progresses and action can become frenetic.

A very good game which gets exciting as the goodies and baddies approach in ever increasing numbers.

Death Trap, on the reverse side of the cassette is a game in which you move your character around the screen in an attempt to avoid being boxed in by the computer. A rather pedestrian game which nevertheless works well.

Dodge-it. Trojan Dragon and Death Trap are all 100% machine code games. The final game on the cassette, Tablets of Hippocrates, is a BASIC program and is a role playing adventure with some 30 locations set in an enchanted forest. Your object is to get the two valuable Tablets hidden deep within the forest. Hazards abound and the forest contains such horrors as a tunnel of death, chamber of echoes and devils rock.

An imaginative game, not unduly difficult and a good introduction to adventuring for the beginner.

Further details on any of the reviews above can be obtained from the following addresses:

Romik Software, 272 Argyll Avenue, Slough, Berks.
Mikro-Gen, 24 Agar Crescent, Bracknell, Berks.
Software Farm, Craigo Farm, Botany Bay, Tintern, Gwent.
Fawkes Computing, 41 Wolfridge Ride, Alveston, Bristol

Spectrum ESP



There have been many unusual applications of the Spectrum but this one from D Barraclough has to be the strangest yet . . .

Everyone has heard of ESP (extra sensory perception) but did you know that scientists investigating the phenomena have categorised it into the following groups:

1 Remote viewing
2 Seeing into the future
3 Telepathy (mind reading)
4 Telekinesis (moving objects by thought)

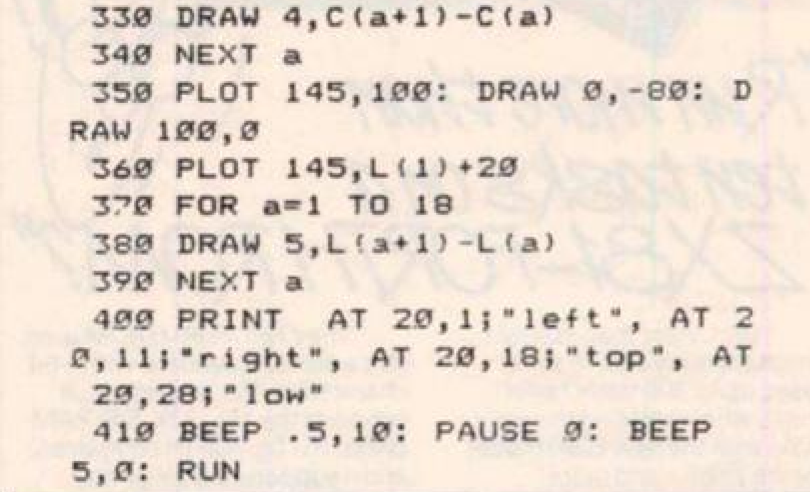
This program is concerned with the last group. Some individuals claim that they can move objects by mental energy alone, and while there has been no conclusive proof documented and published, it is interesting to realise that scientists do not scoff at the idea.

There is a rumour that Atari are being sponsored to develop video games and programs that are thought controlled, the idea being that this will develop a human's mental power — aching brains instead of aching fingers. *ZX Computing* always likes to be among the leaders in any new field and so we present, courtesy of Mr Barraclough, a program to expand the already awesome mental power of our readers.

ESP is a simple program to help you develop any sixth sense which you may possess — it won't give you any if you don't already posess any. In this field you either got it or you aint! The idea is to attempt to keep the moving character in the centre of the screen by your willpower alone. After you have made a lengthy effort — two hours is recommended — pressing any key will produce a graph of the occurances of a central position. This should be repeated daily for best results. The graph shows the number of times the character has appeared at a particular horizontal and vertical position: any tendency for the character to centralise will be displayed.

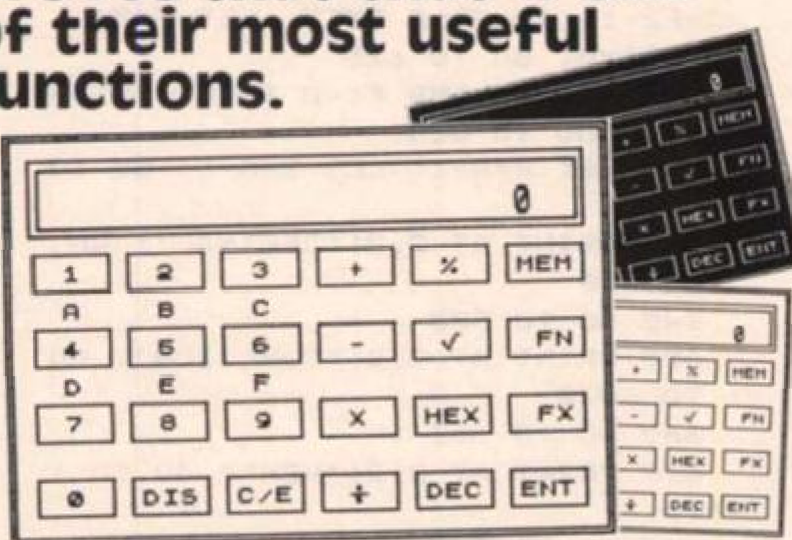
The inspiration for the program apparently came from the Horizons television programme of September 26th 1983 but Mr Barraclough admits that he has not been successful to date. We would be very interested to hear from any reader who gets a favourable result, but no fakers please.

```
  10 BORDER 1: FOR a=0 TO 1: PAP
ER 5: NEXT a
  20 CLS
 100 DIM L(19): DIM C(29)
 110 PLOT 4,4: DRAW 247,0: DRAW
0,168: DRAW -247,0: DRAW 0,-168
 120 LET l= INT ( RND *18)+1
 130 LET c= INT ( RND *28)+1
 140 PRINT  AT l,c; CHR$ 134; CH
R$ 137
 150 PRINT  AT l+1,c; CHR$ 137;
CHR$ 134
 160 LET L(l)=L(l)+1
 170 LET C(c)=C(c)+1
 180 PAUSE 20
 190 PRINT  AT l,c;"  "
 200 PRINT  AT l+1,c;"  "
 210 IF  INKEY$  <> "" THEN GO
TO 225
 220 GO TO 120
 225 PAPER 7: CLS
 230 PRINT  INK 2; INVERSE 1; AT
 1,6;"Occurrence Graphs"
 240 LET sum=0
 250 FOR a=1 TO 19
 260 LET sum=sum+L(a)
 270 NEXT a
 280 PRINT  AT 4,6;"Samples take
n, "; INVERSE 1; INK 3;sum
 290 PRINT  AT 7,2;"Horizontal",
 AT 7,19;"Vertical"
 300 PLOT 10,100: DRAW 0,-80: DR
AW 115,0
 310 PLOT 10,C(1)+20
 320 FOR a=1 TO 28
 330 DRAW 4,C(a+1)-C(a)
 340 NEXT a
 350 PLOT 145,100: DRAW 0,-80: D
RAW 100,0
 360 PLOT 145,L(1)+20
 370 FOR a=1 TO 18
 380 DRAW 5,L(a+1)-L(a)
 390 NEXT a
 400 PRINT  AT 20,1;"left", AT 2
0,11;"right", AT 20,18;"top", AT
 20,28;"low"
 410 BEEP .5,10: PAUSE 0: BEEP
5,0: RUN
```



Cool and calculating

While most of us hotly deny that personal computers are only glorified calculators, B McIntosh of Glasgow proves that that is one of their most useful functions.



This program, for the 16 or 48K Spectrum, is different to other calculator programs in that the functions are displayed as you type them in.

Arithmetic

Enter numbers by pressing the keys as seen. Enter arithmetic functions by pressing the keys on the keyboard where they are printed. For example:

ADD 12+5
PRESS 1,2,K,5,ENT
DISPLAY =17

SUB −13 − 7
PRESS J,1,2,J,7,ENT
DISPLAY = −20

MULT 4 ✱ 32
PRESS 4,B,3,2,ENT
DISPLAY =128

DIV 18 ÷ 6
PRESS 1,8,V,6,ENT
DISPLAY =3

Roots and percent

The square root function is obtained by typing the first letter; ie (R)oot followed by a number. This function can be used on its own or in calculations involving other numbers. For example:

FIND ROOT 2 by typing R,2,ENT
or ROOT 2/2 by typing R,2,V,2,ENT

The percentage function must always have two numbers, one to carry out the function and the multiplier for the percentage. This can also be used on its own or in calculations. For example:

FIND 5% of 17 by typing 1,7,B,5,P, ENT

Memory

The calculator is equipped with eight memories. To store a number there must be an "=" in front of it and then you type "F" to get you into the special function mode followed by "M" for memory and then the number of the memory that you wish to be used. This number can then be used as an operand; ie you can type F,M,1,K,5. This displays M1 +5. Press enter to find the answer. Memories can also be used with each other; ie M1 × M2 etc.

The contents of any memory can be displayed in two ways. Firstly, when the display is showing "0" then by typing as "=10" or such like. Secondly, at any time you can type F,D,M,1. This will display the contents of memory 1 at the top of the display without interfering with any calculations which might be in progress.

The contents of all memories can be reset to zero by typing F,X and following the prompt. If you change your mind, then any other key will abort the memory clear.

Number conversions

This program will also do conversions from HEX to DEC and vice versa. HEX letters are obtained by typing "F" and the number under which they appear. For example:

Find the HEX for 255
Press 2,5,5,H
The display then shows FF

Find the DEC for 1AB
Press 1,F,1,F,2,D
The display then shows 427

The only function left is the C/E which, by typing "C", will erase the last thing that you entered.

Program breakdown

Here follows a brief description of the program functions:

10-40	Sets up variables and flags and puts the computer into CAPS mode.
50-96	Sets up graphics and POKEs the USR routine for the letters into the memory.
98	Screen colour etc.
100-200	Draws the calculator
210-490	More variables
500-560	Handle the input of data (main loop).
600-850	Special instructions for any key pressed which is not a number.
1000-1020	Calculates the answer and running totals.
1100-1140	Converts from decimal to hex.
1200-1300	Displays contents of memory.
1400-1420	Converts from hex to decimal.
1500-1520	Input from keyboard.
6000-6050	Main loop for special functions.
6100-6140	Handles memory functions. Decides whether to update memory or include it in calculations.
6200-6250	Resets all memories to zero.
9000-9060	Converts to and displays large characters.
9900	Saves program.

```
  10 CLEAR 31999: LET H=0: LET Q
=0: LET F=0: LET F1=0: LET F2=0:
 LET F3=0: LET F4=0: LET K=USR "
A"-1152
  15 DEF FN T(G)=G*(E/100)
  20 POKE 23658,8
  30 LET S$="ABCDEFGHIJKLMNOPQRS
TUVWXYZ"
  40 DIM M(8): DIM M$(8,2): LET
M$(1)="M1": LET M$(2)="M2": LET
M$(3)="M3": LET M$(4)="M4": LET
M$(5)="M5": LET M$(6)="M6": LET
M$(7)="M7": LET M$(8)="M8"
  50 LET X=USR "A": FOR N=1 TO 1
6: READ A: POKE X,A: LET X=X+1:
NEXT N
  60 DATA 0,24,24,0,255,0,24,24
  61 DATA 0,3,4,4,200,72,48,16
  70 FOR N=32000 TO 32056
  80 READ A: POKE N,A: NEXT N
  90 DATA 33,0,60,1,0,0,58,57
  91 DATA 125,79,183,22,3,203,33
,203
  92 DATA 16,21,32,249,9,6,4,237
  93 DATA 91,58,125,126,18,19,18
,19
```

```
  94 DATA 35,16,248,237,83,58,12
5,229
  95 DATA 33,Ø,1,25,235,225,6,4
  96 DATA 126,18,19,18,19,35,16,
248,2Ø1
  98 BORDER 7: PAPER 4: INK Ø: C
LS
 1ØØ PLOT Ø,Ø: DRAW Ø,175: DRAW
255,Ø: DRAW Ø,-175: DRAW -255,Ø
 1Ø1 PLOT 2,2: DRAW Ø,171: DRAW
251,Ø: DRAW Ø,-171: DRAW -251,Ø
 11Ø PLOT 1Ø,164: DRAW 235,Ø: DR
AW Ø,-32: DRAW -235,Ø: DRAW Ø,32
 111 PLOT 14,16Ø: DRAW 228,Ø: DR
AW Ø,-25: DRAW -228,Ø: DRAW Ø,25
 12Ø FOR N=1 TO 4
 13Ø FOR L=1 TO 6
 14Ø PLOT L*4Ø-28,N*32-4: DRAW 3
2,Ø: DRAW Ø,-16: DRAW -32,Ø: DRA
W Ø,16
 145 READ A$,B$: PRINT AT N*4+3,
L*5-3;A$;: PRINT AT N*4+5,L*5-3;
B$
 15Ø NEXT L: NEXT N
 16Ø PAPER 7: INK Ø: PRINT AT 2,
2;"                             "
 2ØØ DATA " 1 "," A "," 2 "," B
"," 3 "," C "," + ","," % ","",
"MEM",""," 4 "," D "," 5 "," E "
," 6 "," F "," - ","","    ","","
 FN",""," 7 ",""," 8 ",""," 9 ",
""," X ","","HEX",""," FX","","
Ø ","","DIS","","C/E","","     ","
","DEC","","ENT",""
 21Ø LET A$="Ø": GO SUB 9ØØØ
 22Ø LET Z$="A*CDEFGHI-+L.NO%Q S
TU WXYZ"
 49Ø LET B$="Ø": LET T=Ø
 5ØØ IF A$="" THEN LET A$="Ø":
LET B$=A$
 5Ø1 IF INKEY$<>"" THEN GO TO 5
Ø1
 5Ø2 IF INKEY$="" THEN GO TO 5Ø
2
 5Ø5 BEEP .Ø2,2Ø: PRINT PAPER 7
;AT 2,2;"                    "
 51Ø LET D=PEEK 2356Ø: LET T$=CH
R$ D: LET U$=T$
 515 IF D=13 THEN GO TO 9ØØ
 517 IF T$<>"C" AND T$<>"F" THEN
 LET R=LEN A$: LET S=LEN B$
 52Ø IF D<48 OR D>9Ø THEN GO TO
5ØØ
 53Ø IF D>47 AND D<58 THEN GO T
O 55Ø
 54Ø GO TO 6ØØ+(1Ø*(D-65))
 55Ø IF A$="Ø" THEN LET A$="":
LET B$=""
 554 IF CODE B$<48 AND H=Ø AND C
ODE B$<>Ø THEN LET H=H+1: LET F
1=Ø: LET F2=F2-1
 555 IF F1=Ø THEN LET H=H+1
 56Ø LET F=Ø: LET A$=A$+U$: LET
B$=B$+T$: GO SUB 9ØØØ: GO TO 5ØØ
 6ØØ GO TO 5ØØ
 61Ø LET U$="*": GO TO 1ØØØ
 62Ø IF R=Ø OR S=Ø THEN LET F=Ø
: LET F1=Ø: LET F2=Ø: LET H=Ø: L
ET F3=Ø: LET A$="Ø": LET B$="Ø":
 GO SUB 9ØØØ: GO TO 5ØØ
 622 LET A$=A$(1 TO R): LET B$=B
$(1 TO S): GO SUB 9ØØØ
 623 IF F=1 AND F1=1 THEN LET F
2=F2-1: IF F2=Ø THEN LET F=Ø: L
ET F1=Ø: GO TO 625
 624 IF F=Ø AND F1=Ø THEN LET H
=H-1: GO TO 625
 625 LET R=R-1: LET S=S-1: GO TO
 5ØØ
 63Ø PRINT AT 2,2;"DECIMAL": GO
TO 14ØØ
 64Ø GO TO 5ØØ
 65Ø PRINT AT 2,2;"FUNCTION": GO
 TO 6ØØØ
 66Ø GO TO 5ØØ
 67Ø PRINT AT 2,2;"HEX": GO TO 1
1ØØ
 68Ø GO TO 5ØØ
 69Ø LET U$="-": GO TO 1ØØØ
 7ØØ LET U$="+": GO TO 1ØØØ
 71Ø GO TO 5ØØ
 72Ø LET U$=".": LET T$=Z$(D-64)
: GO TO 554
 73Ø GO TO 5ØØ
 74Ø GO TO 5ØØ
 75Ø LET G=VAL B$(1 TO H): LET E
=VAL B$((H+F2+1) TO ): LET B$=B$
(1 TO H+F2)+"(FN T(G))": LET R=R
-1: LET S=S-1
 755 LET F=Ø: LET A$=A$+Z$(D-64)
: GO SUB 9ØØØ: LET H=Ø: GO TO 5Ø
Ø
 76Ø GO TO 5ØØ
 77Ø LET U$="SQR ": GO TO 1ØØØ
 78Ø GO TO 5ØØ
 79Ø GO TO 5ØØ
 8ØØ GO TO 5ØØ
 81Ø LET U$="/": GO TO 1ØØØ
 82Ø GO TO 5ØØ
 83Ø GO TO 5ØØ
 84Ø GO TO 5ØØ
 85Ø GO TO 5ØØ
 9ØØ IF F=1 THEN GO TO 5ØØ
 91Ø IF A$="Ø" THEN GO SUB 9ØØØ
: LET F3=Ø: GO TO 5ØØ
 915 IF B$="" THEN LET A$="Ø":
 GO SUB 9ØØØ: GO TO 5ØØ
```

```
 920 LET Q=VAL B$: LET A$="="+ST
R$ Q: GO SUB 9000: LET A$="0": L
ET B$=A$: LET F=0: LET F1=0: LET
 F2=0: LET F3=1: LET H=0: FOR N=
1 TO 5: BEEP .01,20: NEXT N
 940 GO TO 500
1000 IF A$="0" THEN  LET A$="":
LET B$="": LET F=1
1002 LET F2=F2+1
1005 IF F1=0 THEN  LET F1=1
1010 IF F=0 THEN  LET Q=VAL B$:
LET A$=STR$ Q: LET H=LEN A$: LET
 B$=A$: LET A$=A$+Z$(D-64): LET
B$=B$+U$: GO SUB 9000: LET F2=1:
 LET F=1: GO TO 500
1020 LET A$=A$+Z$(D-64): LET B$=
B$+U$: GO SUB 9000: GO TO 500
1100 IF F<>0 OR F1<>0 THEN  PRIN
T AT 2,2;"          ": GO TO 500
1105 LET B$="": LET Q=VAL A$: LE
T A=16: LET P=Q
1110 LET Q=P: LET Q=Q-16*INT (Q/
16): LET P=P-Q
1120 LET B$=CHR$ (Q+48+(7*(Q>9))
)+B$
1130 LET P=INT (P/16): IF P<>0 T
HEN  GO TO 1110
1140 LET F=0: LET F1=0: LET F2=0
: LET H=0: LET A$=B$: GO SUB 900
0: LET A$="0": LET B$="0": GO TO
 500
1200 GO SUB 1500
1230 IF W$="M" THEN  PRINT AT 2,
10;"MEM": GO TO 1250
1240 GO TO 1200
1250 GO SUB 1500
1280 IF CODE W$<49 OR CODE W$>56
 THEN  GO TO 1250
1290 PRINT AT 2,14;W$;"=";M((COD
E W$)-48)
1300 LET F=0: GO TO 500
1400 IF F<>0 OR F1<>0 THEN  PRIN
T AT 2,2;"          ": GO TO 500
1410 LET B$="": LET Q=0: FOR N=1
 TO LEN A$: LET P=CODE A$(N)-48-
(7*(CODE A$(N)>64))
1420 LET Q=Q*16: LET Q=Q+P: NEXT
 N: LET A$=STR$ Q: GO SUB 9000:
LET A$="0": LET B$="0": GO TO 50
0
1500 IF INKEY$<>"" THEN  GO TO 1
500
1510 IF INKEY$="" THEN  GO TO 15
10
1520 LET W$=CHR$ PEEK 23560: BEE
P .02,20: RETURN
6000 GO SUB 1500
6005 IF A$="0" THEN  LET A$="":
LET B$=A$
6010 IF W$="X" THEN  GO TO 6200
6020 IF W$="M" THEN  GO TO 6100
6030 IF W$="D" THEN  PRINT AT 2,
2;"DISPLAY  ": GO TO 1200
6040 IF CODE W$>47 AND CODE W$<5
5 THEN  LET T$=CHR$ (CODE W$+16)
: LET A$=A$+T$: GO SUB 9000: GO
TO 500
6050 GO TO 6000
6100 PRINT AT 2,10;" MEM"
6101 GO SUB 1500
6104 IF CODE W$<49 OR CODE W$>56
 THEN  GO TO 6101
6110 PRINT AT 2,15;W$
6120 LET V=CODE W$-48
6125 IF F=0 THEN  LET A$="": LET
 B$=""
6127 IF F3=0 THEN  LET A$=A$+M$(
V): LET B$=B$+STR$ M(V): LET H=L
EN B$: GO SUB 9000: LET F=0: LET
 F1=0: LET F3=1: LET F4=1: GO TO
 500
6130 IF F<>0 THEN  LET A$=A$+M$(
V): LET B$=B$+STR$ M(V): LET T$=
"": LET F=0: GO SUB 9000: GO TO
500
6140 LET M(V)=Q: PRINT AT 2,2;"
                  ": LET T$="": LET
 A$="": GO TO 500
6200 PRINT AT 2,2;"MEMORY CLEAR-
PRESS X"
6210 GO SUB 1500
6230 IF W$<>"X" THEN  GO TO 6250
6240 DIM M(8): PRINT AT 2,2;"
                      ": GO TO 500
6250 PRINT AT 2,2;"
      ": GO TO 500
9000 PRINT AT 3,2;"
             ";AT 4,2;"
"
9005 POKE 32058,60: POKE 32059,1
25
9010 FOR N=1 TO LEN A$
9020 IF CODE A$(N)>90 THEN  POKE
 32001,K-256*INT (K/256): POKE 3
2002,INT (K/256)
9025 POKE 32057,CODE A$(N)
9030 RANDOMIZE USR 32000
9040 POKE 32001,0: POKE 32002,60
: NEXT N
9050 POKE 23606,52: POKE 23607,1
23: PRINT  PAPER 7; INK 0;AT 3,2
8-(LEN A$);S$(1 TO LEN A$): POKE
 23606,60: POKE 23607,124: PRINT
 PAPER 7; INK 0;AT 4,28-(LEN A$
);S$(1 TO LEN A$)
9060 POKE 23606,0: POKE 23607,60
: RETURN
9900 SAVE "CALCULATOR" LINE 1
```

Little Brothers should be seen but not heard.



A maxim which eloquently describes the Brother HR-5.

Less than a foot across, it's nonetheless loaded with features.

But there's one thing the HR-5 won't give you. Earache.

For the annoying 'clickety clack' many printers produce is mercifully absent from the HR-5.

Quietly efficient, it delivers high definition dot matrix text over 80 columns at 30 c.p.s.

The HR-5 also has something of an artistic bent.

Being capable of producing uni-directional graph and chart images together with bi-directional text.

It will also hone down characters into a condensed face, or extend them for added emphasis.

Incorporating either a Centronics parallel or RS-232C interface, the HR-5 is compatible with most home computers and popular software.

Perfectly portable, the battery or mains operated HR-5 weighs less than 4lbs, and has a starting price of only £179.95 (inc.VAT).

Which is really something to shout about.

PLEASE SEND ME MORE DETAILS OF THE REMARKABLE BROTHER HR-5 PRINTER.

NAME________________

ADDRESS________________

ZXC6/84

TEL.NO.________

brother

DEPT P, BROTHER OFFICE EQUIPMENT DIVISION, JONES + BROTHER, SHEPLEY STREET, GUIDE BRIDGE, AUDENSHAW, MANCHESTER M34 5JD

1K disassembler

Disassemble your ZX81 ROM in just 1K — sounds impossible? Well, have a go with this program written for us by HM Tucker of Luton.

This program is designed to decode any area of ROM or RAM into instructions comprising op-code and data if any, and to print them out with their addresses, one to a line, until the screen is full (approximately one second in Fast mode) and groups all Z80 instructions.

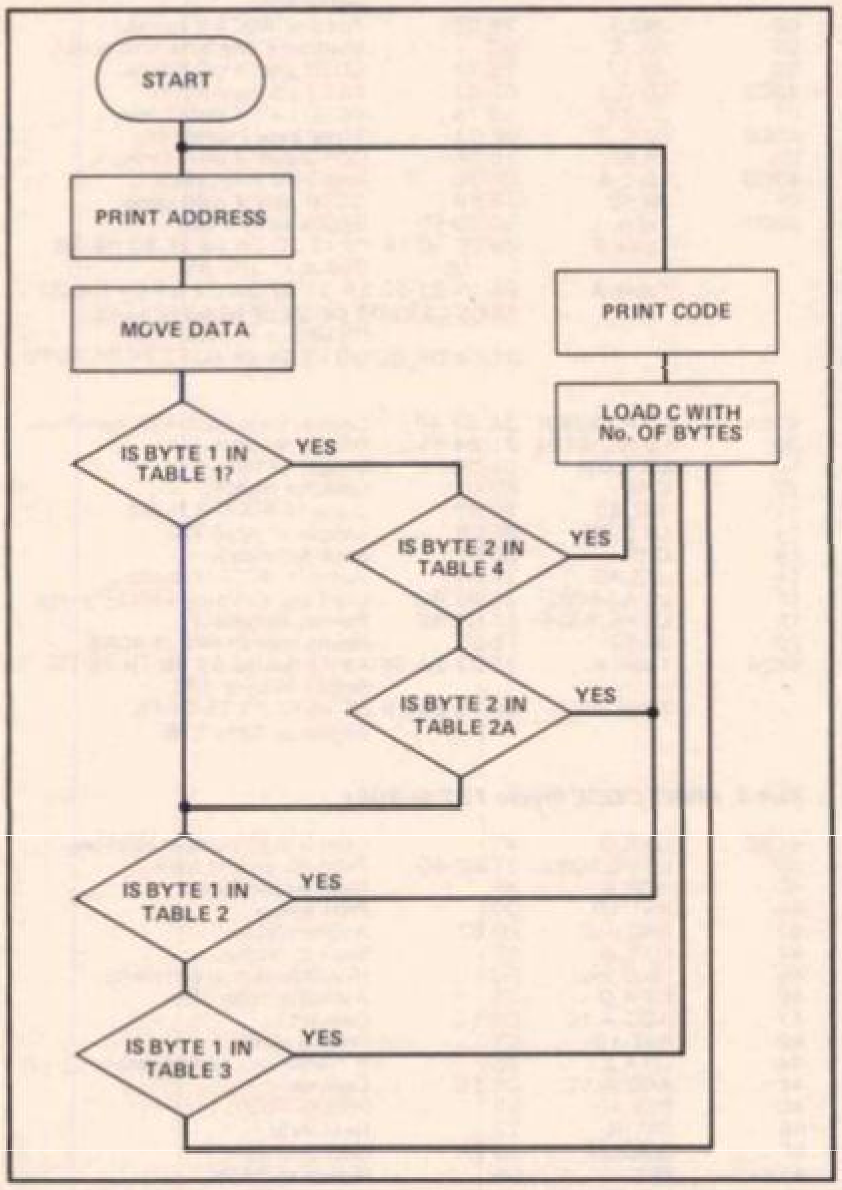
It fits easily into an unexpanded ZX81, and will also run in Slow mode, which is essential if you wish to change the program to scroll so that you may sit back and watch the entire ROM unfold before you (about six instructions per second in Slow mode).

It accepts up to four digits of Hex address, but if decimal addressing is preferred, simply change lines 10 and 20 and omit lines 30 to 60. Options are available for copying to printer, continuing disassembly or of selecting new start address.

Apart from being very interesting to write and to run, it will allow you to find the misprints in some published versions of what is in the ROM.

How it does it

The BASIC program asks for the start address in Hex; this is converted to decimal by lines 30 to 60 and POKEd into spare locations in the system variables area by lines 70 and 80.





The machine code now takes over and *Part 1* prints out the address as four digit Hex. The program doesn't seem to have achieved much so far does it? But these few bytes can be tucked away above RAMtop or in the printer buffer as a handy decimal to hex converter.

Part 2 moves four bytes of data from the start address to the four reserved bytes at the start of the REM ready for *Part 3* to look at, and for *Part 4* to print as necessary.

Part 3 is the longest and most complicated of the whole program and comprises all the decision blocks in the flow chart, and all the tables. Briefly, it searches the tables guided by a pointer in HL and a length in BC, looking for a match for a particular byte of the op-code. The accumulator is loaded with the first byte and table 1 is searched; *if a match is not found in table 1 all* we can be sure of, is that we do not have a four byte instruction, but more about this later.

Tables two and three are then searched for two and three byte instructions respectively; any code escaping these searches must of necessity be a one byte instruction. All these instructions have one byte op-codes

```
LOADER

ENTER NEW CODE FOR CHANGES

USE "Q" TO QUIT

USE NEWLINE TO ACCEPT CODE

   1 REM .........0.........0...
......0.........0.........0.....
....0.........0.........0.......
..0.........0.........0.........
0.........0.........0.........0.
........0.........0.........0...
......0.........0........
  10 LET A=16513
  20 LET B=A/A
  30 LET T=B
  40 IF B=209 THEN GOTO 240
  50 GOSUB 200
  60 PRINT AT 5,0;B;" IS ";H$;"?
";
  70 INPUT H$
  80 IF H$="" THEN GOSUB 200
  90 IF H$="Q" THEN STOP
 100 IF LEN H$<>2 THEN GOTO 70
 110 LET D=CODE H$*16+CODE H$(2)
-476
 120 IF D>255 THEN GOTO 70
 130 POKE A+B,D
 140 LET T=T+D
 150 PRINT TAB (10);H$
 160 SCROLL
 170 LET B=B+1
 180 GOTO 40
 200 LET D=PEEK (A+B)
 210 LET I=INT (D/16)
 220 LET H$=CHR$ (I+28)+CHR$ (D-
I*16+28)
 230 RETURN
 240 PRINT "CHECKSUM = ";T
```

and in all cases C is loaded with the appropriate value and control passed to *Part 4* to print code.

If a match *was* found in table 1 we may have an instruction two, three or four bytes long, it depends on the value in byte two, so the accumulator is loaded with the second byte and table 4 searches for four byte instructions then table 2A for two byte instructions beginning with DD or FD. Again C is loaded with the appropriate value and control passed to *Part 4* to print code.

I told you it was complicated didn't I? Don't worry though, the worst is yet to come. We are now left with 50 two byte instructions which began with ED, and 48 three byte instructions which began with DD or FD. Nobody wants 98 bytes of tables, so the accumulator is re-loaded with the first byte and the main program rejoined at table 2 where the 50 two byters are picked out by ED. And so to table 3 where the 48 three byters are picked out by DD and FD. Again C is loaded and control passed to *Part 4*.

Part 4 prints out the appropriate number of bytes and is virtually the same as *Part 1*, but this time prints a space before each byte to make the display easier to read. Since we don't care about corrupting the data, this time we can leave out the rotate left digit necessary in *Part 1*. On return to BASIC at line 100, the decimal address is updated by the number of bytes used and the screen examined to see if it is full. If not then back to line 70, and off we go again until it is full.

Loading the REM

This is without doubt the worst part of all, but the following hints may help. Use Fast mode to load the REM with 208 characters, eg a sequence of nine full stops and a zero repeated makes counting quite easy, but as a check when you have finished, a direct command of PRINT PEEK 16511 should give a value of 210.

Use any loader program you may have that loads Hex or use the following which although rather long and with limited protection on the input, is easy to use. When RUN, it will display the current contents of the first byte of the REM as '1 is 1B?' and wait for

1. Print address (bytes one to 29).

ADDRESS	MNEMONIC	CODE	REMARKS
4082	4xNOP	00 00 00 00	Watch this space.
86	LD B,2	06 02	Counter for two byte address.
88	XOR A	AF	Clear accumulator.
89	LD HL,407C	21 7C 40	Address of hi byte of start.
8C	RRD (HL)	ED 67	Move nibbles.
8E	LD E,A	5F	Save lo nibble.
8F	LD D,(HL)	56	Hi nibble into accumulator avoiding code 7E.
90	LD A,D	7A	
91	ADD A,1C	C6 1C	Convert.
93	RST 10	D7	Print hi nibble.
94	LD A,E	7B	Lo nibble into accumulator.
95	ADD A,1C	C6 1C	Convert.
97	RST 10	D7	Print lo nibble.
98	SUB A,1C	D6 1C	Restore the byte in (HL).
9A	RLD,(HL)	ED 6F	
9C	DEC,HL	2B	Address of lo byte of start.
9D	DJNZ,ED	10 ED	Jump to 408C.

Part 2. MOVE DATA (bytes 30 to 40)

9F	LD BC,4	01 04 00	Byte counter.
A2	LD DE,4082	11 82 40	Destination, is the reserved space.
A5	LD HL,(407B)	2A 7B 40	Source, is start, or updated address.
A8	LDIR	ED B0	Move it.

Part 3. SEARCH (bytes 41 to 186)

40AA	LD A,(4082)	3A 82 40	Load accumulator with byte one.
AD	LD HL,40D1	21 D1 40	Pointer to table 1.
B0	LD C,3	0E 03	Length of table 1.
B2	CPIR	ED B1	Look for match.
B4	JRZ,54	28 54	Jump to 410A if found, to look at byte two.
40B6	LD C,1A	0E 1A	Length of table 2.
B8	CPIR	ED B1	Look for match (two byte instruction).
BA	JRZ,9	28 09	Jump to 40C5 if found.
BC	LD C,1C	0E 1C	Length of table 3.
BE	CPIR	ED B1	Look for match (three byte instruction).
C0	JRZ,7	28 07	Jump to 40C9 if found.
C2	INC C	0C	Must be a one byte instruction.
C3	JR,77	18 77	GOTO *part 4* print code.
40C5	LD C,2	0E 02	Two byte instruction.
C7	JR,73	18 73	GOTO *part 4* print code.
40C9	LD C,3	0E 03	Three byte instruction.
CB	JR,6F	18 6F	GOTO *part 4* print code.
40CD	LD C,4	0E 04	Four byte instruction.
CF	JR,6B	18 6B	GOTO *part 4* print code.
40D1	Table 1	DD ED FD	Begins at byte 80.
	Table 2	06 0E 10 16 18 1E 20 26 28 2E 30 36 38	Begins at byte 83.
	Table 3	01 11 21 22 2A 31 32 3A C2 C3 C4 CA CC 3E C6 CB CE D3 D6 DB DE E6 ED EE F6 FE CD	Begins at byte 109.
		D2 D4 DA DC DD E2 E4 EA EC F2 F4 FA FC FD	
410A	LD A,(4083)	3A 83 40	Load accumulator with byte two.
0D	LD HL, 4124	21 24 41	Pointer to table 4.
10	LD C,0D	0E 0D	Length of table 4.
12	CPIR	ED B1	Look for match.
14	JRZ,B7	28 B7	Jump to 40CD if found.
16	LD C,0B	0E 0B	Length of table 2A.
18	CPIR	ED B1	Look for match.
1A	JRZ,A9	28 A9	Jump to 40C5 if found.
1C	LD A,(4082)	3A 82 40	Load accumulator with byte one.
1F	LD HL,40D4	21 D4 40	Pointer to table 2.
22	JR,92	18 92	Rejoin mainstream at 40B6.
4124	Table 4	21 22 2A 36 43 4B 53 5B 63 6B 73 7B CB	Begins at byte 163.
	Table 2A	09 19 23 29 2B 39 E1 E3 E5 E9 F9	Begins at byte 176.

Part 4. PRINT CODE (bytes 187 to 208)

413C	LD B,C	41	Load B with number of bytes.
3D	LD HL,4082	21 82 40	Point HL at first byte.
40	XOR A	AF	Clear accumulator.
41	RST 10	D7	Print space.
42	RRD,(HL)	ED 67	Rotate nibbles.
44	LD E,A	5F	Save lo nibble.
45	LD D,(HL)	56	Hi nibble into accumulator.
46	LD A,D	7A	Avoiding code 7E.
47	ADD A,1C	C6 1C	Convert.
49	RST 10	D7	Print hi nibble.
4A	LD A,E	7B	lo nibble into accumulator.
4B	ADD A,1C	C6 1C	Convert.
4D	RST 10	D7	Print lo nibble.
4E	INC HL	23	Next byte.
4F	DJNZ,EF	10 EF	Until B is zero.
4151	RET	C9	Return to BASIC.

BASIC PROGRAM

```
 10 PRINT "START IN HEX."
 20 INPUT H$
 30 LET S=0
 40 FOR N=1 TO LEN H$
 50 LET S=S*16+CODE H$(N)-28
 60 NEXT N
 70 POKE 16508,INT (S/256)
 80 POKE 16507,S-PEEK 16508*256
 90 LET N=USR 16518
100 LET S=S+N
110 PRINT
120 IF PEEK 16442=2 THEN STOP
130 GOTO 70
```

you to either accept it by pressing Newline, or to change it by entering the desired code followed by Newline.

In either case the code will be POKEd in and displayed to the right of the old code. The next byte is then displayed below the first, so carry on and enter all 208 bytes, and good luck, remember to start with four 'no ops'.

If you make an error leave it for corrections later, the byte numbers will help you keep in step with the program which will stop after 208 bytes and display a simple checksum which should be 22880; if it isn't you have made an error. Run the loader again and step through till you find it. Note that a simple checksum is not an absolute guarantee of accuracy but you may proceed with fair hope of success. Save it onto tape just in case. Delete all the loader except the REM in line 1 starting with the last line and working back to line 10.

Basically . . .

Enter the BASIC program, select Fast mode, enter RUN Newline and you should see: 'START? IN HEX'.

Type in a start address, say 0, and with your fingers crossed press Newline; all being well you should have no more than one second of suspense followed by a display of Hex addresses and groups of code, which are the first 22 instructions in ROM. The program will stop with error code 9/120.

You may now COPY to printer if required, continue disassembly by CONT Newline or use RUN Newline to select a new start address.

Assuming all has gone according to plan, it is now play time; why not look at our own program using 4086, then use 4082 and note that even though not starting at the beginning of an instruction, that the program is self aligning. Instead of the boring old THEN STOP in line 120, try THEN CLS or THEN SCROLL but do select Slow mode before running these last two.

Notes

There are seven possible routes through the flow chart corresponding to the following table:

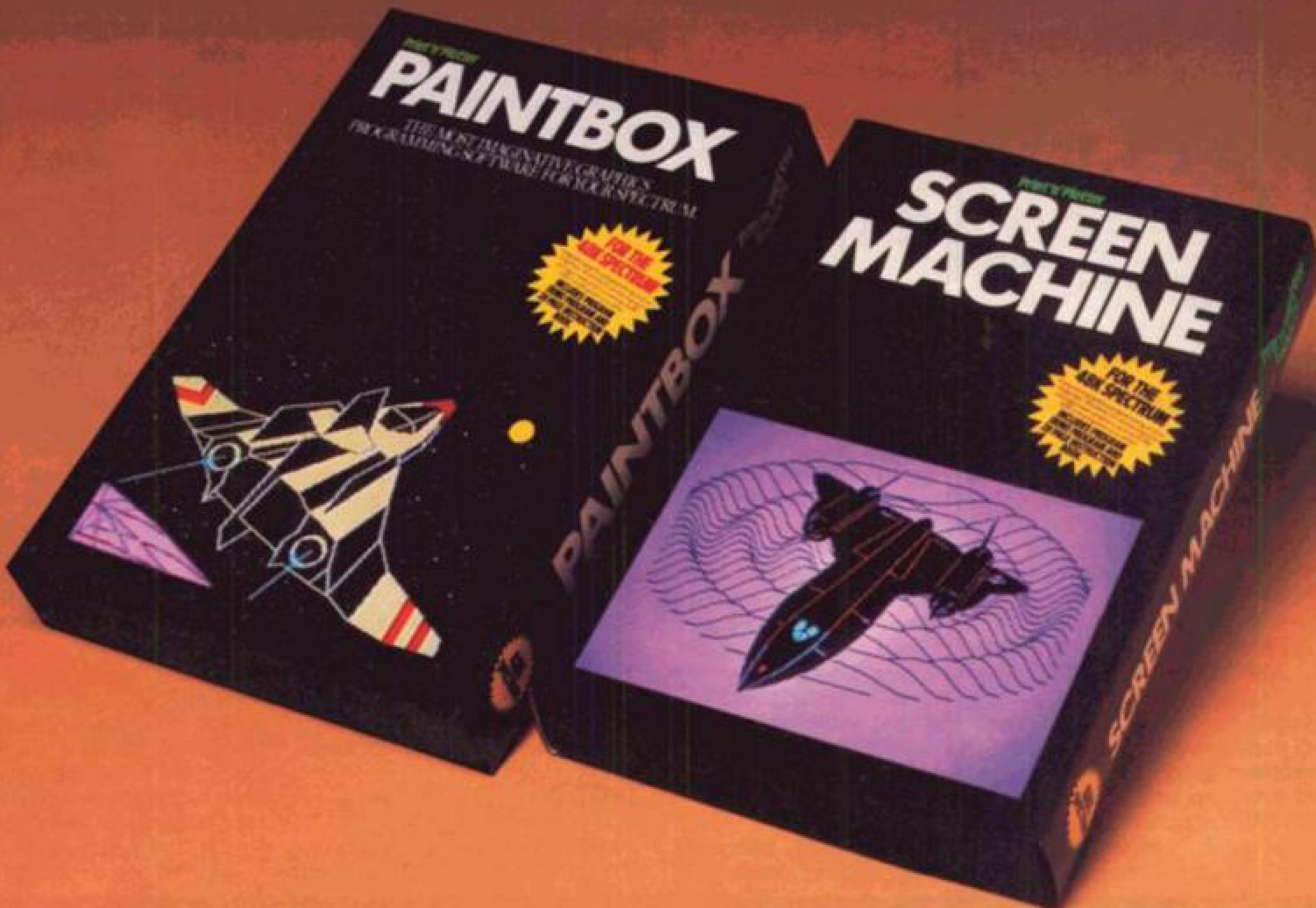
number of bytes		
opcode	data	instruction
1	0	1
1	1	2
1	2	3
2	0	2
2	1	3
2	2	4
3	1	4

For our purpose it has been sufficient to look no further than the second byte of the op-code, since all 62 three byte op-codes form four byte instructions which are detected by their second byte.

In the interests of easy programming and description, no great attempt has been made to save memory, since even at the most crowded part of the ROM around 0650 there is plenty of room on the screen for all the code.

My thanks to Dr Ian Logan's book 'Understanding your ZX81 ROM' for the inspiration to write this program. Should you require more information on the techniques used in my program, I suggest you look no further than this excellent book.

The Software.



The Hardware.



The Facts.

PRINT 'N' PLOTTER

PAINTBOX

Superb Graphics – without expensive hardware!

PAINTBOX gives the 48K Spectrum owner *every* facility to produce excellent graphics — easily, quickly and professionally.

There's no need for expensive hardware. PAINTBOX is flexible, precise and incredibly simple to use!

Just take a look at some of the facilities available:

UDG EDITOR

The facility to define up to 84 graphics characters which can be stored in your BASIC programs for instant recall from its own built-in machine code!

UDG DRAWING BOARD

UDG planner for the 4 banks of characters. Facilities include: MIRROR IMAGE, ROTATE, INVERSE, FILE.

SKETCHPAD

Experimentation 'window' that allows you to try-out your UDG ideas under development.

PRECISION PLOTTER

Amazing versatile high-resolution drawing board which includes PAPER choice, PLOT, DRAW, DRAW RADIALLY, CIRCLE, ARC, OVER, FILL, instant change of INK and BRIGHT, multi-function ERASE and STORE.

Cursor is controlled by keyboard or Joystick with FAST, SLOW, and Cursor size choice!

SCREEN PLANNER

For the best of both worlds! PRECISION PLOTTER and UDG CHARACTERS! For complete screen planning of graphics. A multi-purpose facility to enable you to produce superb screen graphics!

All results can be sent to a printer, saved as SCREEN$ or SAVED as CODE with its built-in machine code routine for instant recall from BASIC.

The program is complete with a cassette demo and a 28 page booklet describing in easy-to-understand language how to use it, plus many tips for storing and using your graphics in programs.

PRINT 'N' PLOTTER

SCREEN MACHINE

Instant machine code for graphics and text.

SCREEN MACHINE is an invaluable graphics utility to use with PAINTBOX or any other graphics hardware or software!

It gives a wide range of facilities to enable you to manipulate graphics and text, saving time, memory and giving truly professional results.

For instance: SCREEN MACHINE COMPRESSES Screen Files to cram even more into the memory!

It COMPILES text, graphics characters or UDG's instantly into machine code giving fast smooth access to the screen!

It CREATES re-callable Multiple Screen Files automatically with its own built-in machine code!

It ENLARGES screen sections in 2X steps!

It REDUCES the whole screen in 2X steps!

It RECOLOURS your screen graphics either globally or selectively!

It MIRRORS the screen left to right making flip-action animation easy!

It RELOCATES your screen graphics to any other part of the screen with block or high-resolution scrolls!

It SUPERIMPOSES screen graphics giving incredible 'mixing' results!

It SAVES everything to tape or Microdrive for use within your BASIC programs!

There's no doubt that SCREEN MACHINE is a major graphics toolkit for the 48K Spectrum. It is completely menu-driven and comes complete with demo program and fully-documented instruction booklet.

Another first from Print 'n' Plotter!

PRINT 'N' PLOTTER

ZX SPECTRUM JOTTER

Pre-planning your screen made easy . . . and precise!

Print 'n' Plotter JOTTERS have become a household word for the Sinclair enthusiast.

Despite various imitations our original ZX SPECTRUM JOTTER is still the one people prefer!

Of course it could be because it is professionally produced . . . the quality is superb.

And the fact that it is BIG SIZE A3 (16½" x 11¾") is a distinct advantage when working in high-resolution.

It's also 100 pages thick. 50 pages of PLOT grids showing each numbered pixel co-ordinate and 50 pages of PRINT grids showing every character and graphic character position and INPUT lines.

Each page also contains 24 UDG planning grids (2400 per pad).

Consider also the fact that it is printed on Artist's Detail paper . . . thick enough to take any writing, drawing or colouring, yet thin enough to overlay onto a drawing and trace-off.

For pre-planning graphics, text, tabulation or anything to produce 'on screen' a Print 'n' Plotter JOTTER won't be beaten.

The complete package comes with a set of coloured pens, a Pixel ruler and a handy corrugated storage tray.

If you use PAINTBOX, SCREEN MACHINE or any other graphics utility . . . you'll do things better with the ORIGINAL Print 'n' Plotter JOTTER!

PRINT 'N' PLOTTER

KEYBOARD OVERLAYS

The simple answer to "Which key?"

Print 'n' Plotter KEYBOARD OVERLAYS for the standard ZX Spectrum keyboard are the economic answer to "Which key does what?"

If you program, or buy commercially-produced software, sooner or later you'll be faced with a mind-boggling mass of keys that perform different functions.

This can be as simple as the "FIRE" button in a game or as precise as the "analysis" button in a scientific program.

Let's face it, the Spectrum keyboard is complicated enough so why not take the easy way to remember . . .

. . . lay over the keyboard a Print 'n' Plotter OVERLAY and write the function underneath . . . it's child's play!

Then keep the OVERLAY for the next time you use the program.

Print 'n' Plotter KEYBOARD OVERLAYS come in packs of ten. Punched to fit your Spectrum. Priced to suit your pocket!

PRINT 'N' PLOTTER

ZX PRINTER PAPER

Print 'n' Plotter Products also market ZX PRINTER PAPER that gives good, clear print, eminent feeding qualities – probably the best you'll find in the U.K.

FAST DELIVERY and economically priced . . . you won't find better!

CREDIT CARD 24HR. ANSAPHONE ORDERS

01-660 7231

(Phone 01-403 6644 for Dealer and other enquiries.)

Post to: Dept ZX Print 'n' Plotter Products Ltd., 19 Borough High Street, London SE1 9SE.

Please send me:

.... "PAINTBOX" SOFTWARE @ £8.25

.... "SCREEN MACHINE" SOFTWARE @ £8.25

.... SPECTRUM JOTTER PADS @ £9.00

.... SPECTRUM KEYBOARD OVERLAYS @ £2.95

.... 5 ROLLS ZX PRINTER PAPER @ £12.50

All prices include U.K. postage and VAT.

☐ I enclose remittance in full

☐ Please bill my Access/Barclaycard/Visa/Mastercard No:

Overseas orders please add 25% on quoted prices, for additional handling.

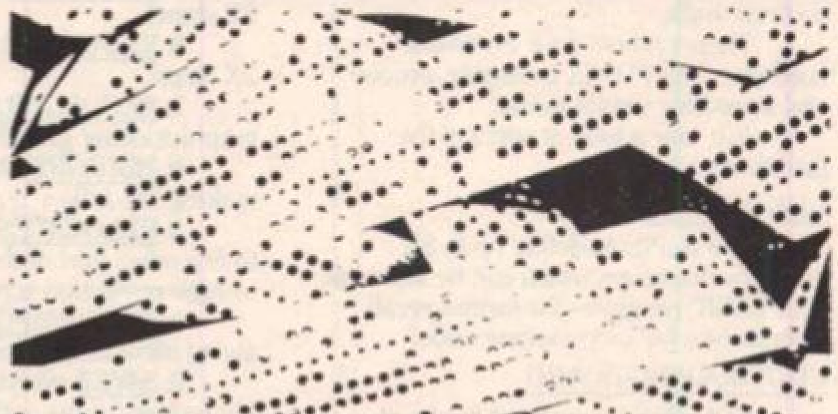
NAME____________________

ADDRESS____________________

Print 'n' Plotter Products

Data handling

Keith Williams of Newport Pagnell gives us some 'pointers' on this oft needed subject.



In my job as a teacher I have, often, to handle large quantities of data which needs to be sorted in a number of different ways. This is the sort of task for which a computer is ideally suited, but how do you write the program to perform these tasks?

Let us take an example. The example that I shall use is one from teaching but one could very easily think of equivalent examples in business, or sports or anywhere else that some form of data base is used.

In this example I will consider a year group of 120 pupils in four classes. They are studying six subjects (Maths, English, Computer Studies, History, Geography and Science). These are referred to by their initial letters. The information I need to store (and sort) is:

Name, Class, Age (dob), Results in M,E,C,H,G,S

So far, this is easy. I can set up a character array A$(120,50) and put in the information for each child. As parents rarely consider how many letters there are in the name that they give to their child, the amount of string space taken up by the name will vary. Still, this is easily overcome by taking a maximum name length of, say, 30 characters and using Sinclair's "Procrustean" slicing-off of feet or padding with spaces to fit the name into spaces 1 to 30. The rest of the data can be put into other specific areas of the string, eg class will be held in A$(x, 31 to 33).

This is still very simple. But the whole aim of the exercise is not just to hold the data but to be able to use it. I need to be able to sort the list in alphabetical order throughout the year and within each class, in exam result order for each subject and for the total mark overall both within each class and across the whole year. Why is this so difficult? Fig 1 can be used to illustrate this.

If they are sorted in alphabetical order then A$(1) refers to Fred, A$(2) to John and A$(3) to Harold. If I then want to sort according to Maths marks, then A$(2) has to become A$(1), A$(3) becomes A$(2) and so on. If I now want to sort according to English or History or anything else, then the strings have to play musical chairs. Large amounts of data have to be moved about everytime sorting, resorting and sorting again. Moving large amounts of data is slow and inefficient. Therefore, it is bad programming practice. What is needed is a more efficient way of linking and handling the data.

Data stacks

This is where planned data structures come in. If we call all the data relating to one child a record, then this is divided into a number of fields as shown in Fig 2.

As you can see, each field is represented by a different array. The common index i links them all. So, for i = 1 then N$(i) is Fred Bloggs, M(i) is his maths mark and so on. The importance of the pointer array will become apparent soon. The value of i will vary from 1 to the total number of students. This can be input when the program is first run. This number held as a variable, say total, can be used to DIM the arrays. If the initial DIM statements are of the form DIM N$(total+5,30) then we can hold 30 character long names for each pupil and allow for five new pupils to join the school during the year.

The pointer is used to give the index of the next record in the list. Two other variables are needed, toplist and topempty. When the program is first run the data stack will look like Fig 3a. Topempty tells me the index of the first empty record, P(i) tells me the index of the next empty record and so on through the chain. When I input data for my first record this will go into the one indicated by i = 1. Toplist will now hold 1, P(I) will become 0 and topempty will become 2. After 3 record entries my data stack will now look like Fig 3b.

	NAME	MATHS MARK	ENGLISH MARK	HISTORY MARK
A$(1)	BLOGGS FRED	30	80	56
A$(2)	COLLINS JOHN	60	27	30
A$(3)	SMITH HAROLD	45	42	70

Fig 1

INDEX	NAME	CLASS	d.o.b.	MATHS	ENGLISH	C.S.	HIST	GEOG.	SCI.	POINTER
i	N$(i)	C$(i)	D$(i)	M(i)	E(i)	C(i)	H(i)	G(i)	S(i)	P(i)

Fig 2

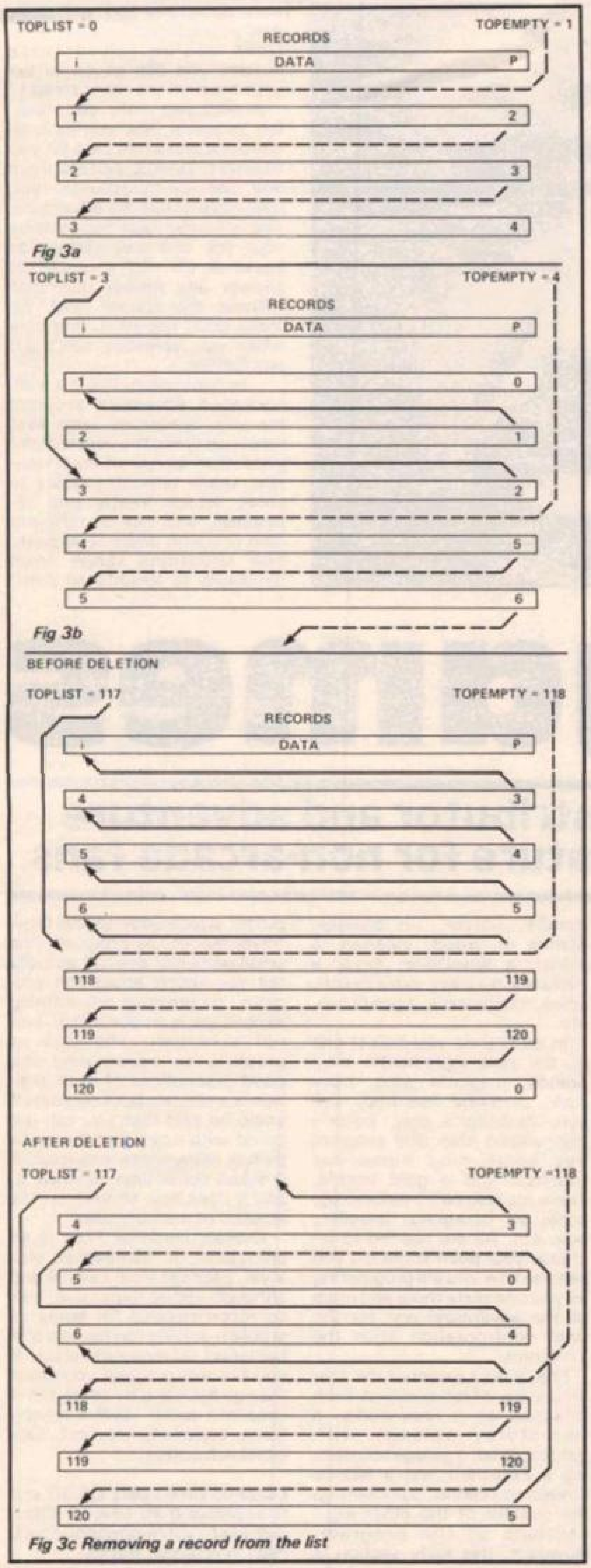


Fig 3a

Fig 3b

Fig 3c Removing a record from the list

Adding and removing records

If a child leaves then I don't even have to delete his record. I just add it to the empty list (on the end) and adjust the relevant pointers. Fig 3c shows what happens to the data stack when the student whose record is held in 5 leaves the school. Note that no data has been moved, just three point values changed. Similarly if a new pupil arrives, he is assigned the data lock pointed to by the variable topempty and then this block is tied on to the full chain in the relevant position. Again three pointer values altered and no data moved.

Sorting

What about sorting? My original problem was how to sort the data in n different ways efficiently and without moving masses of it around. Well, this is where my rather complex method of storing data comes in to its own. Let us take the simplest case of only sorting by one criterion, eg alphabetical order.

Using the original model we would compare two strings. If one was 'greater' than the other then they would swap places using some such lines of BASIC as:

```
1000 If A$ (i) > A$ (j) THEN LET
B$ = A$ (j):LET A$ (j) = A$ (i):
LET A$ (i) = B$
1010 NEXT j: NEXT i
```

On our model we would move just two numbers not massive data blocks:

```
1000 IF A$ (i) > A$ (j) THEN LET
temp = P(j): LET P(j) = P(i) : LET
P(i) = temp
1010 NEXT j : NEXT i
```

But this is not the end of the story. If P(i) is the pointer in sorting by alphabetical order, we can use other pointers to sort by all other criteria. For example, r(i) could be the pointer for sorting by Maths mark, q(i) for sorting by English mark and so on. We would need new variables to act as pointer to the top of each list, eg topmaths, tophist etc.

The sort routines can be as above so that if m(i) > m(j) then r(i) and r(j) would swap values.

To print names in order of maths mark we would use a routine such as:

```
5000    POKE 23658,8
5010    INPUT " Hard copy or
        to screen (H/S)"; LINE
        B$
5020    IF B$ (I) = "H"
        THEN OPEN #2,"p"
5030    LET i = topmaths
5040    IF i = 0 THEN RETURN
5050    PRINT N$ (i)
5060    LET i = r(i)
5070    GOTO 5040
```

Line 5000 locks on CAPS SHIFT (see *ZX COMPUTING* Vol 1 No 8 p 126) Lines 5010 and 5020 direct output to printer or screen. 5030 print the names held in the topmaths-r(i) pointer chain in the sorted order.

Final tidying up

All that really remains now is to tidy up the many arrays and then sit down to coding.

The neatest way to hold all the pointers and numerical data is in one three-dimensional array. If we do this in the array a() then it would be DIMmed a(m,2,i).

The first dimension — m — represents the number of fields which will be stored or sorted (don't forget to allow for the string arrays). The second dimension — 2 — represents pointer or data. The numerical data held in 1 and the pointers held in 2. Finally, the third dimension — i — holds the actual data and pointers (i here is the index that we have used throughout).

If maths, for example, were held as field no 3, then the maths sorting routine described above would be written comparing a(3,1,i) with a(3,1,j), if it is larger then we would swap pointers a(3,2,i) and a(3,2,j)

Only two routines are needed to sort the data — a string sort routine and a numerical sort. First a string array is set up to hold the names of the fields — say S$ — so that S$ (2) holds ENGLISH, S$ (3) holds MATHS and so on. Remember to hold these as capitals and to convert all input into capitals. Then a simple FOR — NEXT loop will find the correct field:

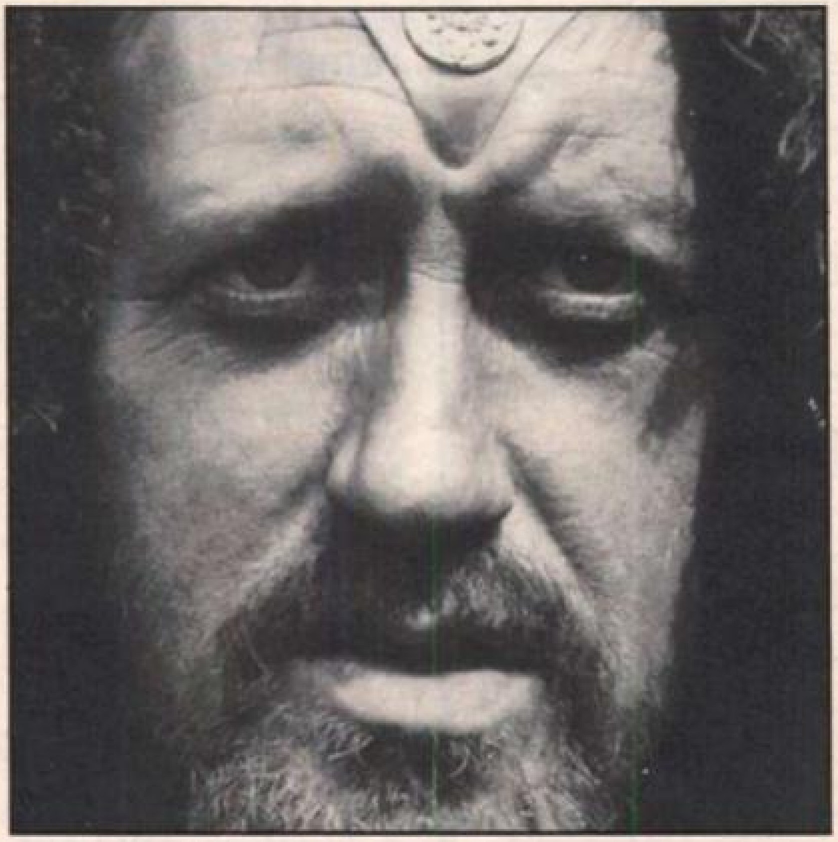
```
8050 INPUT "Which field do
you want to sort?" ; LINE B$
8060 FOR n = 1 TO Number of
fields
8070 IF S$ (n,1 TO 3) = B$ (1
TO3) THEN GOTO number sort
8200 REM number sort
8205 IF a(n,1,i) > a(n,1,j)
. . . . . and so on as before
```

The routine above can also be used to automatically direct operation to the string sort routine. Similarly, the print subroutine can use Ss.

A data saving and loading facility needs to be written to make the data base program complete: now all that is left is the coding!

Level 9 are well known for their adventure programs. Their latest offering, Lords of Time, follows hot on the heels of the Middle Earth trilogy (Colossal Adventure, Adventure Quest and Dungeon Adventure) and the Silicon Dream trilogy (Snowball, Return to Eden and The Worm in Paradise). Lords of Time is written in what is called "a super-compact adventure language; a-code". This gives fast response times with long text messages but doesn't occupy much memory.

The tape comes in an attractive (if superfluous) large package and has a brief ten-page pamphlet (like Dark Crystal) which contains highly cryptic clues in the form of an introduction and a poem. The tape appears to be well duplicated and loads first time. Although it only occupies 32K of memory, the program is very wide in scope, having some 300 locations, 750 messages, and about 80 objects. The objective is to



very absorbing and will easily hold your interest for some time. Once you've solved these puzzles you can go on to tax your brain on the other zones.

Should you really get stuck, the program box contains an envelope and a hint card for you to send to Level 9, and ask them any specific question(s) you may have about the adventure. This is a nice touch, but could be your one and only chance as Level 9 do not promise to answer any further questions without the special card. So make good use of it, and only when you definitely can't get any further.

In conclusion, this is a well-packaged adventure program for the Spectrum with fast response times to avoid tedium, good descriptions of each location, many difficult puzzles to solve which keeps you interested, and lots of different (and unusual) areas to explore. The situations range from dinosaurs to knights to com-

Mindgames

Greg Turnbull, regular contributor and adventure fanatic, begins a regular feature for non-arcade fans.

score points by collecting each of 9 ingredients and 18 treasures.

Bonus points are awarded for certain actions and for completing the game; points are lost if you get killed! The central location point is the clock from which nine time-zones can be reached. These must be completed in numerical order, as objects from previous zones are required to solve problems in subsequent ones. The ultimate aim of the adventure is to defeat the evil Timelords.

The program recognizes most of the usual adventure words such as: examine inv(entory), score (out of 1,000), drop, look, give, take, N, S, SE, NW, U, D, forward, backward, enter, get eve(rything), search, open, wear, it, etc. AGAIN repeats the last command but there is no HELP routine. The vocabulary understands some 200 words and will accept short sentences, eg search the bedroom, examine the clock, etc. However, you may need to rephrase some requests that it doesn't understand.

The usual quit/save game features are available — the word RESTORE will load a previously saved situation. This feature is always welcome if you get stuck and want to go away and think about a problem over a cup of tea. The speed of reply is excellent and the scenes described are very detailed. However, no graphics are available as this is an old-style adventure without character interaction and not in 'real-time'. Hence it suffers badly in comparison to the likes of The Hobbit and Valhalla — perhaps the closest equivalent would be Artic's series of text-only adventures.

Some of the problems are very perplexing; one of the earliest being how to get to each of the time-zones, and later how to get the maddeningly-elusive keys under the door of the garden shed! Each of the time-zones contains various locations and problems for you to solve. Numerous objects are provided to help you in these tasks, eg a rope, spear, tin of catfood, metronome, a pick, matchbox, candle, mirror, tin opener, planks of wood, valerian (a plant), a lodestone, keys, a rucksack (to carry extra items), coins, mushrooms, a petrol can, etc.

In each zone you collect one of the nine ingredients: olive branch, dragon's wing, ivory tusk, diamond teardrop, evil eye, dinosaur's egg, jester's cap, silicon chip (the program has some nice humorous touches) and a gold buckle. There are also two treasures per zone, an hourglass, jewellery box, etc. All are needed to increase your point score (so you can see how you are progressing as you complete more and more of the adventure) and for the final confrontation with the Timelords.

I have only explored the first time-zone which contains such locations as a road-works, a two-storey cottage with garden, shed, a garage (containing a Porsche!) and a nearby stream so I cannot comment on the content of the other eight sections of the program. However, this early section is

puters which adds to the time-travel feel of the program. The small pamphlet doesn't actually tell you much about the program, so previous adventuring experience is an advantage (but not essential). The lack of graphics to accompany the good descriptions of each location is a shame, but I suppose it could be said that you can get bored with looking at the same picture many times (especially if it takes some time to draw it) and it does help to save on the amount of memory used.

Overall, Lords of Time is an enjoyable, if somewhat old-style, package from Level 9 and although rather expensive, can be recommended for hours of problem-solving fun for new and hardened adventurers alike. If you like this program you could then go back and try some of the excellent earlier Level 9 adventures, especially the first, Colossal Adventure.

Lords of Time costs £9.90 and is available from Level 9 Computing at 229 Hughendon Road, High Wycombe, Bucks.

De-bugger

Getting a program typed in is often only the start of your problems. Ed to the rescue.

Typing in a program is a useful exercise. Apart from the patience required, techniques learned and the end program to be used, probably the most educational part of it is tracking down the bugs introduced by yourself or occasionally by our publication system.

In debugging you gain a much deeper insight and understanding on how the program actually works than by merely typing it in, but tracking down these errors is an art in itself and needs some skill. So here are some tips to help you in your efforts when faced with that cryptic error report!



1 NEXT without FOR
Look back through the program, either the loop has not been set up — no related FOR 'letter' = No1 TO No2 line, or the letter has been re-used as an ordinary variable within the loop with a LET 'letter' = No.

2 Variable not found
This is one of the most common errors. Again, the problem may not lie in the line where the error was detected and reported. If there is only one variable, which may be one or more letters or a string ($) variable, then that is the problem. There may be more than one variable in the line section reported and you will have to identify the offending one. In a line PRINT AT Y,X;A$ the culprit could be Y or X or A$. To find out which of them is causing the problem (it may be more than one) type in turn as a direct command:

PRINT Y Enter/Newline
PRINT X Enter/Newline
PRINT A$ Enter/Newline

Note which produces the error report. Now look back through the program printout for the line which sets it up — usually a LET or FOR command. Did you leave it out? Does the program get there or has a GOTO/GOSUB been wrongly addressed?

3 Subscript wrong
Connected with DIM A(No) or DIM A$(No). If the number in the brackets on the line where the error is reported is greater than the one in the original DIM statement, is not an integer or is less than 1, then this report is generated. If the subscript — number in brackets — is a number then check and change, however, if it is a variable then follow the procedure for tracing variables. It has probably exceeded the limits, look for lines with the variable being altered with + − * / : if necessary add limiting code. For example:

IF X >10 THEN LET X = 10

4 Out of memory
As well as for programs which are too big, it may happen if the previous program set RAMtop. Before despairing, enter CLEAR USR "a"-1 on the Spectrum: on the ZX81 SAVE the program, turn the machine off and on, then reload the program.

7 RETURN without GOSUB
Somehow the computer has reached a RETURN command other than via a GOSUB instruction. Check a GOTO hasn't been entered in place of a GOSUB. Check for a missing GOSUB.

B Integer out of range
An integer (whole number) either as a number of variable is too big or small and you are attempting to do something like PRINT AT 0,33 — not allowed! Check any variables involved as per report 2 and trace it back through the program looking for adjustments to it by + − * / : Add limiting code if needed — see report 3

E Out of DATA
A Spectrum problem. Check the number of DATA items match the number of READs; usually one (or more) has been missed out. Attempting to reread a DATA list without first using a RESTORE command will cause this and it can happen on an auto start program (saved with a LINE number). Good programming usually RESTOREs to the correct line number before using READ.

I FOR without NEXT
See report 1 but this time the NEXT is missing!

Note that the letters I have used for examples could by ANY letters not just A$,X,Y etc and depend on the particular choice of the programmer.

This is by no means a comprehensive list but I have tried to cover many of the most common error reports. Personally, I get almost as much satisfaction from debugging as I do from programming I do assure you, however, that there is absolutely no truth in the rumour that we deliberately inject bugs into our listings in order to introduce you to the dubious delights of debugging!

Conversion tips

A guide to ZX81/Spectrum program conversions from David Nowotnik.

The versions of BASIC offered by the two ZX computers are so similar that many programs for one can be used by the other. The ZX81 has only two commands which are not present on the Spectrum, SCROLL and UNPLOT, and these should cause you few problems when converting ZX81 programs to the Spectrum (see Table 1).

There are quite a lot of commands and functions on the Spectrum which are not available on the ZX81. A list of these appears in Table 4. The stars indicate those commands and functions for which there is no simple translation to ZX81 BASIC. Those for colour and sound can be ommitted; but you will have to find some alternative for the high resolution and file I/O commands.

The command PLOT appears on both computers, but the effect is quite different, so beware! Another tip: PEEK and POKE should be used with caution. In conversion, addresses will almost certainly have to be changed. Some of those changes appear in the tables. A command such as POKE USR "a" . . . . on the Spectrum indicates User Defined Graphics; ZX81 users don't have this facility, so you'll have to omit this and use a standard character instead.

ZX81	Spectrum	Comments
SCROLL	RANDOMISE USR 3582 or LET t=USR 3582	If the program uses random numbers, they could become rather predictable with the first option. If so, use the second, using a variable (in this case t) which is otherwise not used.
PLOT Y,X	PRINT AT 21−Y/2,X/2;	Print the appropriate quarter square graphics character.
UNPLOT Y,X	PRINT AT 21−Y/2,X/2;	Print a space, or the appropriate quarter square graphics character.

Table 1 ZX81 to Spectrum conversions.

Spectrum	ZX81	Comments
BIN eg LET y=BIN 10010101	LET y=(decimal no.) Conversion to decimal: 10010101=149 128 64 32 16 8 4 2 1 Add these numbers together when a 1 appears at the appropriate position in binary.	BIN allows the representation of a number in binary. On the ZX81 use the decimal equivalent, but beware; BIN is often used with User Defined Graphics, which are not available on the ZX81.
READ/DATA eg READ x,y DATA 50,60	LET LET X=50 LET Y=60	READ and DATA are used to store a lot of information in a program. Use LET instead.
DEF FN and FN eg DEF a(x)=SQR x LET t=FN a(i)	LET X$="SQR X" LET X=I LET T= VAL X$	The defined function can appear in a string. Use the keyword for built-in functions (eg SQR). The equivalent of FN may need 2 lines, as shown.
PLOT	no equivalent	
SCREEN$ eg LET a=SCREEN$ x,y	LET A=PEEK(PEEK 16396 +256*PEEK 16397+1+Y+33*X)	Used in interactive games to detect characters in the display file. Note — this formula only works when a RAM pack is fitted.

Table 2 Spectrum to ZX81 conversions.

ZX81	Spectrum	Comments
1 FRAMES POKE 16436,255 POKE 16437,255	POKE 23672,0:POKE 23673,0	Both computers have a counter which accurately varies by 50 every second. In the example, use the first line to start the 'clock'. The variable T will have the time in seconds after the start. The counter can only be used for 10 minutes.
LET T=(65535-PEEK 16436-256*PEEK 16437)/50	LET t=(PEEK 23672+256*PEEK 23673)/50 For times greater than 10 minutes, you can use byte 23674 as well.	
2 Line number zero POKE 16510,0	POKE 23756,0 (As the start of BASIC can move, eg with microdrives) use with caution.	Converts the first line of a program to line number zero, which cannot be edited, and so is protected.
3 RAMTOP POKE 16388,X-256*INT (X/256) POKE 16389, INT (X/256)	CLEAR x	Creates a safe area at the top of RAM starting at address x, for storing data, machine code etc.

Table 3 General interconversion hints.

BEEP	*	FORMAT	*	ATTR	*
BORDER	*	INK	*	BIN	
BRIGHT	*	INVERSE		FN	
CAT	*	MERGE	*	IN	*
CIRCLE	*	MOVE	*	OVER	*
CLOSE	*	OPEN	*	POINT	*
DATA		OUT	*	SCREEN$	
DEF FN		PAPER	*	VAL$	*
DRAW	*	READ			
ERASE	*	RESTORE	*		
FLASH	*	VERIFY	*		

Table 4 Spectrum functions not available on the ZX81.

Currency conversion

If you're going abroad this summer, this program from K J Rider in Kingston is a must!

Currency Conversion will store and retrieve the rates of exchange between stirling and any other currency of a wide range of countries.

Keith has used some neat programming techniques to fit it into 16K, one of which is holding the rates of exchange in a REM statement and using the PEEK/POKE functions to store and retrieve it. The program even allows for lucky microdrive users by using the system variables to get the 1 REM address.

When first run you will have to use the option to set up each of the exchange rates (these can be found at any bank) and the program will crash if they are not entered before attempting to convert currencies. Countries other than those supplied in the listing may be changed or added to and the following notes will help you to make any adjustments.

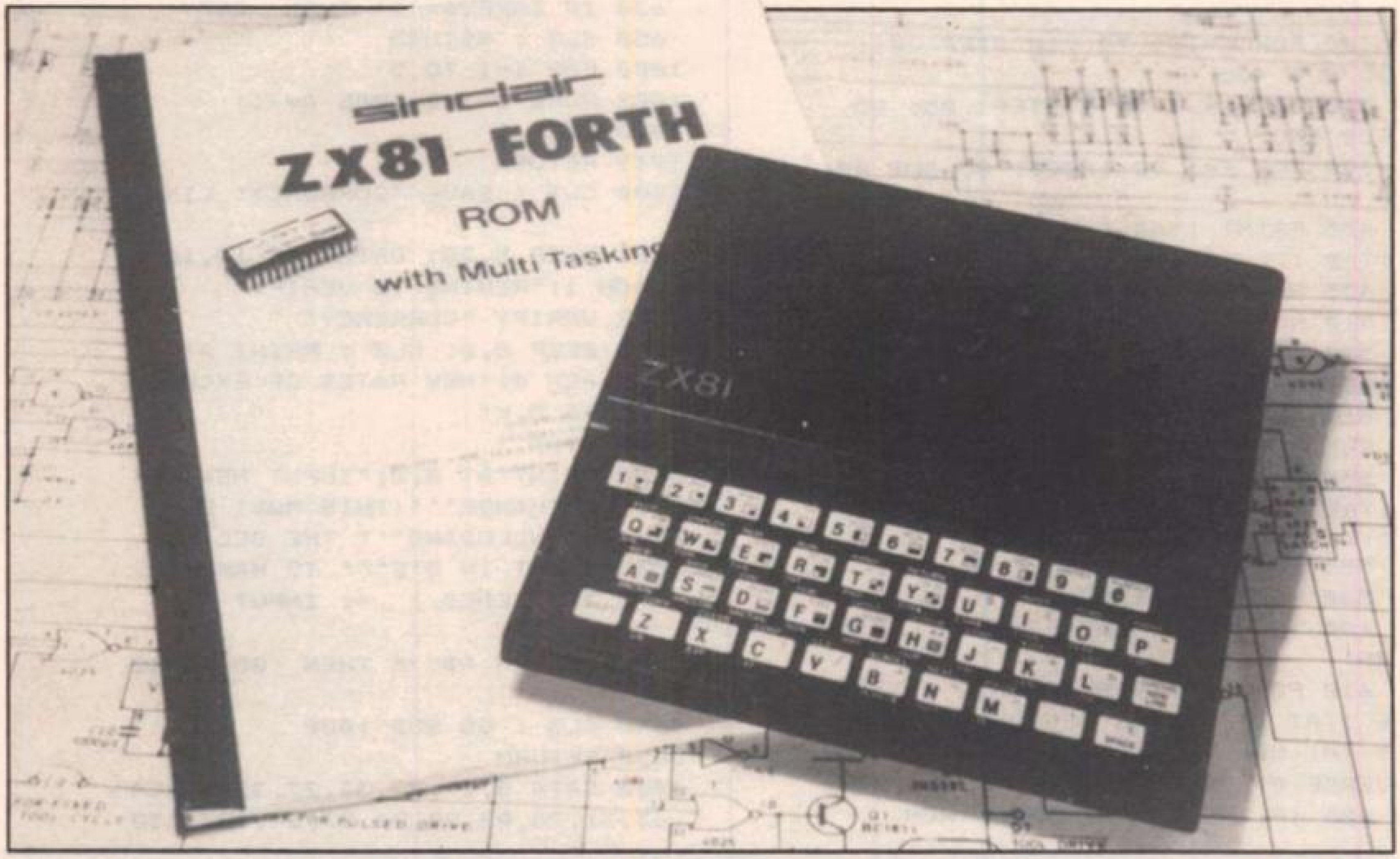
Program details

1	REM statement used to store the currency conversion rates
20-30	Strings containing various instructions and information
35	Sets CAPS SHIFT on the Spectrum
40-45	Prints titles
50	String with countries and currencies, modify for other places but watch spacing and positioning.
60-69	Prints Menu
70-80	SAVE or get rate of exchange
90-650	Display routines
1000-15	POKE new rates to place in REM
1500/05	SAVE routines
5000/50	Get new rates routines
6000	DATA of B$ offsets for each country used in printing the correct country and currency in line 90

Bon voyage!

```
   1 REM ........................
.................................
.................................
   5 CLS
  10 REM CURRENCY CONVERSION BY
K.J.RIDER
  20 LET S$="?                PR
ESS (P)"
  30 LET G$="THERE ARE- ": LET M
$=" TO THE POUNDS": LET O$="DO Y
OU WISH TO CHANGE THE RATE ?  (Y
)-YES   (N)-NO."
  35 POKE 23658,8
  40 PRINT "  CURRENCY RATES FOR
 THE POUND"'"___________________
______________"''" SELECT THE COU
NTRY AND PRESS "'"THE KEY INDICA
TED."
  45 PRINT ';TAB 3;"COUNTRY__";T
AB 12;"CURRENCY"
  50 LET B$="AUSTRIA SCHILLINGS
BELGIUM FRANCS FRANCE GERMANY MA
RKS HOLLAND GUILDERS ITALY LIRA
SPAIN PESETAS SWITZERLAND UNITED
 STATES DOLLARS"
  60 PRINT ';TAB 3;B$( TO 19);TA
B 28;"(1)"
  61 PRINT ;TAB 3;B$(20 TO 34);T
AB 28;"(2)"
  62 PRINT ;TAB 3;B$(35 TO 40);B
$(27 TO 34);TAB 28;"(3)"
  63 PRINT ;TAB 3;B$(42 TO 54);T
AB 28;"(4)"
  64 PRINT ;TAB 3;B$(56 TO 72);T
AB 28;"(5)"
  65 PRINT ;TAB 3;B$(73 TO 83);T
AB 28;"(6)"
  66 PRINT ;TAB 3;B$(84 TO 96);T
AB 28;"(7)"
  67 PRINT ;TAB 3;B$(98 TO 109);
B$(27 TO 34);;TAB 28;"(8)"
  68 PRINT ;TAB 3;B$(110 TO );TA
B 28;"(9)"
  69 PRINT ';"PRESS  (S) TO SAVE
 THE"'" PROGRAM WITH THE NEW RAT
ES"'"OF EXCHANGE."
  70 PAUSE 0
  75 LET Z$=INKEY$: IF (Z$<"1" O
R Z$>"9") AND Z$<>"S" THEN  GO T
O 75
  80 CLS
  85 IF Z$="S" THEN  GO TO 1500
  87 LET Z=PEEK 23635+256*PEEK 2
3636-4+10*VAL Z$: LET D$="": FOR
 I=Z TO Z+4: LET D$=D$+CHR$ PEEK
 I: NEXT I
  90 RESTORE : FOR I=1 TO 2*VAL
Z$ STEP 2: READ A,B: NEXT I
  92 PRINT G$;D$;B$(A TO B): PRI
NT M$( TO 13): PRINT 'O$
  95 PRINT 'O$( TO 22): PRINT B$
(A TO B);M$( TO 4);M$(8 TO );S$:
 PAUSE 0: CLS
  96 IF INKEY$="Y" THEN  GO SUB
5000: GO TO 87
 104 IF INKEY$="P" THEN  GO TO 1
06
 105 CLS : PRINT ;TAB 4;M$(8 TO
);TAB 14;B$(A TO B): GO TO 290
 106 PRINT ;TAB 4;B$(A TO B);TAB
 16;M$(8 TO ): GO TO 500
 290 PRINT "______________________
```

```
-----------"
 300 FOR K=.01 TO .10 STEP .01:
GO SUB 400
 310 FOR K=.1 TO 1 STEP .05: GO
SUB 400
 320 FOR K=1 TO 10000: GO SUB 40
0
 400 PRINT ;TAB 5;K;TAB 12;" = "
;TAB 17;K*VAL D$
 405 GO SUB 600
 410 NEXT K: RETURN
 500 FOR K=.1 TO 1 STEP .05: GO
SUB 550
 510 FOR K=1 TO 10000 STEP 1: GO
 SUB 550
 550 PRINT ;TAB 5;K;TAB 12;" = "
;TAB 17;(INT (((K*(1/VAL D$))*10
0)+.5)/100)
 555 GO SUB 600
 560 NEXT K: RETURN
 600 IF PEEK 23689<>2 THEN  RETU
RN
 610 POKE 23659,0: PRINT  INVERS
E 1;AT 22,0;" (S)-SCROLL (C)-COP
Y (M)-MENU    ": POKE 23659,2: IN
VERSE 0: PAUSE 0
 625 IF INKEY$="M" THEN  RUN
 630 IF INKEY$="C" THEN  COPY
 650 CLS : RETURN
1000 FOR I=1 TO 5
1005 POKE Z-1+I,CODE A$(I)
1010 NEXT I
1015 RETURN
1500 CLS : SAVE "CURRENCY" LINE
1
1501 BEEP 5,20: PRINT AT 10,10;
FLASH 1;"REWIND TO VERIFY"
1502 VERIFY "CURRENCY"
1503 BEEP 8,0: CLS : PRINT AT 0,
0; FLASH 0;"NEW RATES OF EXCHANG
E SAVED O.K"
1505 STOP
5000 PRINT AT 0,0;"INPUT NEW RAT
E OF EXCHANGE"'"(THIS MUST BE 5
DIGITS INCLUDING"'" THE DECIMAL
POINT. PUT IN 0'S"'" TO MAKE UP
TO 5 IF NEEDED.)   ": INPUT "
 ";A$
5020 IF LEN A$<>5 THEN  GO TO 50
00
5040 CLS : GO SUB 1000
5050 RETURN
6000 DATA 8,19,27,33,27,33,49,54
,63,71,78,82,89,96,27,34,123,130
```



The ZX81 goes Forth

Peter Shaw tries an unusual variation on an old favourite.

When the Ed told me about a ZX81 with a Forth ROM the same thoughts flashed across my mind as when I heard someone had linked the '81 to a 800K disc drive and an Epson printer: for what possible reason would they go to the trouble? Well, although I have the same feelings about the drive and printer, Forth on the '81 is a slightly better idea.

The EPROM is either sold as a single complete unit, ie a ZX81 with a Forth EPROM fitted, or as a 'diy' conversion kit, so that you can take your existing ZX81 and exchange the BASIC ROM for Forth.

Fitting the chip is quite simple, if your BASIC ROM is in a holder. All you have to do is whip one out with a screwdriver, and drop in the new one. If your BASIC ROM is soldered in then you'll probably be better off just buying the complete Forth package. David Husband, the producer of ZX81-Forth, calmly tells you to de-solder the basic ROM in one line of the manual. Taking one of those ROMs out is not very easy for one thing, and even if you do manage to get it out, you'll probably never be able to use it again and you might have damaged some of the other chips on the PCB while using the soldering iron.

Once you've managed to get your EPROM in and working, what have you got? Well, it's not the Fig-Forth that I know and love, but what it loses on the swings, it gains on the roundabouts. ZX81-Forth is fast. I mean FAST, even compared to other versions of Forth it is pretty nippy. In AUTO (FAST mode), the Forth version of a FOR-NEXT from 1 to 30000 takes less than 1 second. One major feature of Forth is that it is 'Multitasking' this means ZX81 Forth was probably developed with real-time controlling in mind. There are no graphics, as such, so the games players amongst you will be disappointed if you think this will provide a simple way of writing games without resorting to machine code.

I couldn't find any bugs in the system while I was playing with the machine, this seems too good to be true!

Conclusions

As ZX81-Forth is based on TREE-FORTH and not the popular Fig Forth, I am not as thrilled with ZX81-Forth as I could have been. There must be a very limited market for ZX81 owners who want to run their hot-water boiler with a Forth program written on their small Sinclair. Apart from that, ZX81-Forth is a well written ROM.

Addendum: Since Peter wrote this review I have had many conversations with suppliers of control units who all said that the ZX81 has taken on a new lease of life with Forth as a control unit — one telling me of a University which has installed several to monitor experiments.

Although, as Peter says, perhaps the home user may have a limited, specialised use for it, I find it interesting to know of the applications for which the humble home computer is being used. If you know of any special ways in which these machines are being used, why not write in and let me know.

Spectrum owners are not forgotten as David Husband also supplies a FORTH ROM unit for this computer. **RE**

ZX81 Forth costs £25 plus £2 p&p. (VAT is not included.) It is available as an Eprom, or complete ZX81 with Eprom fitted from: David Husband, 2 Gorleston Road, Branksome, Poole, Dorset BH12 1NW.

Snake Trap

Don't get hissterical over this chase game from Pete Cooke in Leicester!

In this program, your snake (the grey squares) has to try and trap the computer's snake (inverse 'c's). You can move using the arrowed keys 5 to 8.

The squares occupied by the snakes are contained in two 1 dimensional arrays, A(50) and B(10). For the player's snake, two pointers, AA and AB, move by POKEing in the new head at A(AA) and wiping out the tail at A(AB). Changing (AA-AB) lengthens or shortens the snake.

As the computer's snake is always ten squares long, it only needs one pointer, BB moving through the array. The old value of B(BB) is overwritten and the new value replaces it.

Line by line

Here follows a breakdown of the program which should help you understand the listing a little better:

90-660	Forms the game loop. The player's move is input and the computer line 150 checks that it is to an empty square. Line 130 allows for the player's snake finding food. Lines 300-550 check for the computer's snake. If the next square is a wall then it is either right or left at random.
1000-2050	Shows who has won the game and reruns the program for another game.
5000-6990	Sets up the variables and gives instructions.
7000-7200	Sets up the board.

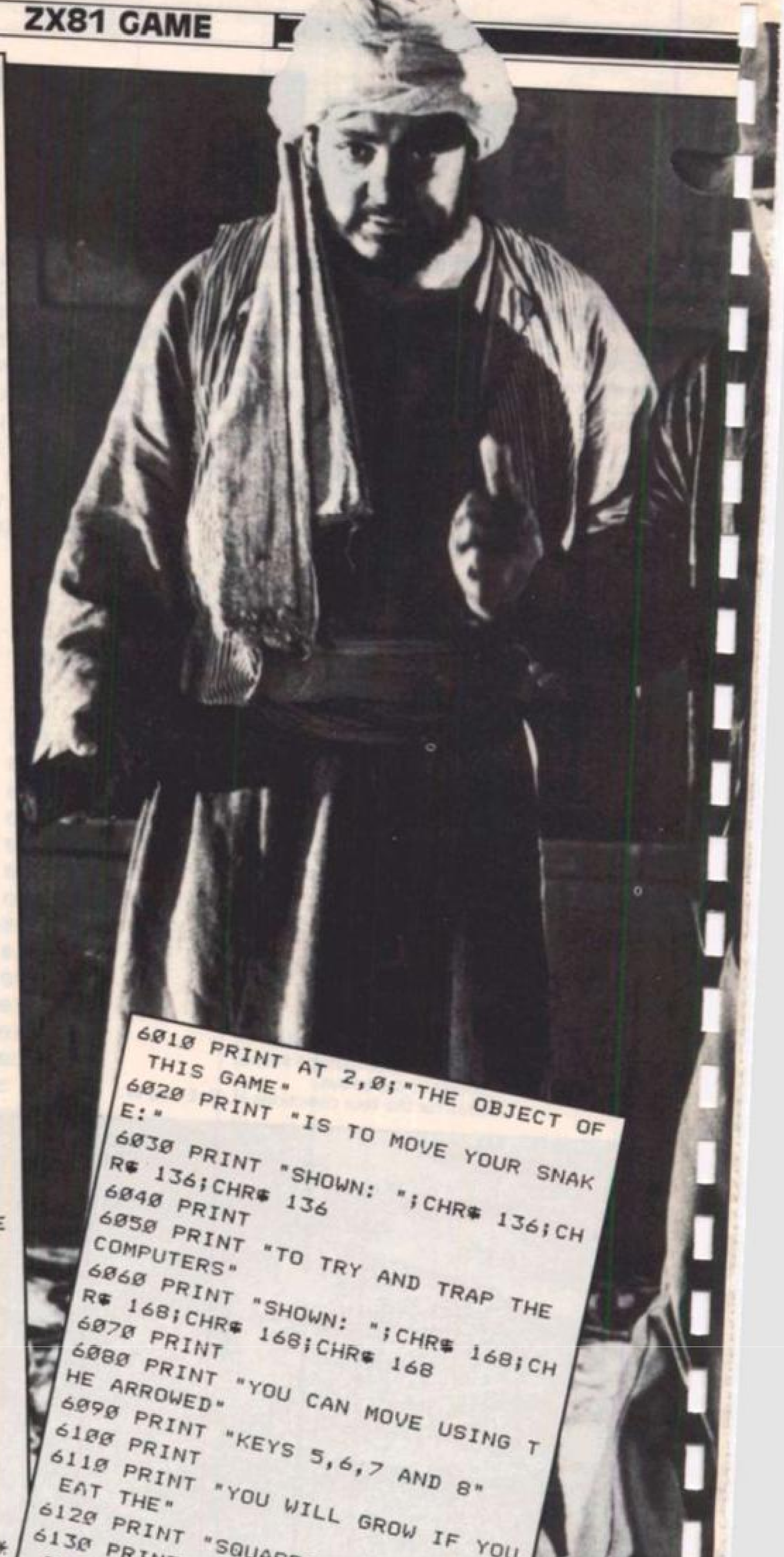
Variables

A(50)	Positions of player's snake.
B(10)	Positions of computer's snake.
AA,AB,BB	Pointers in the arrays.
P(4)	Codes for the four directions N, S, E and W.

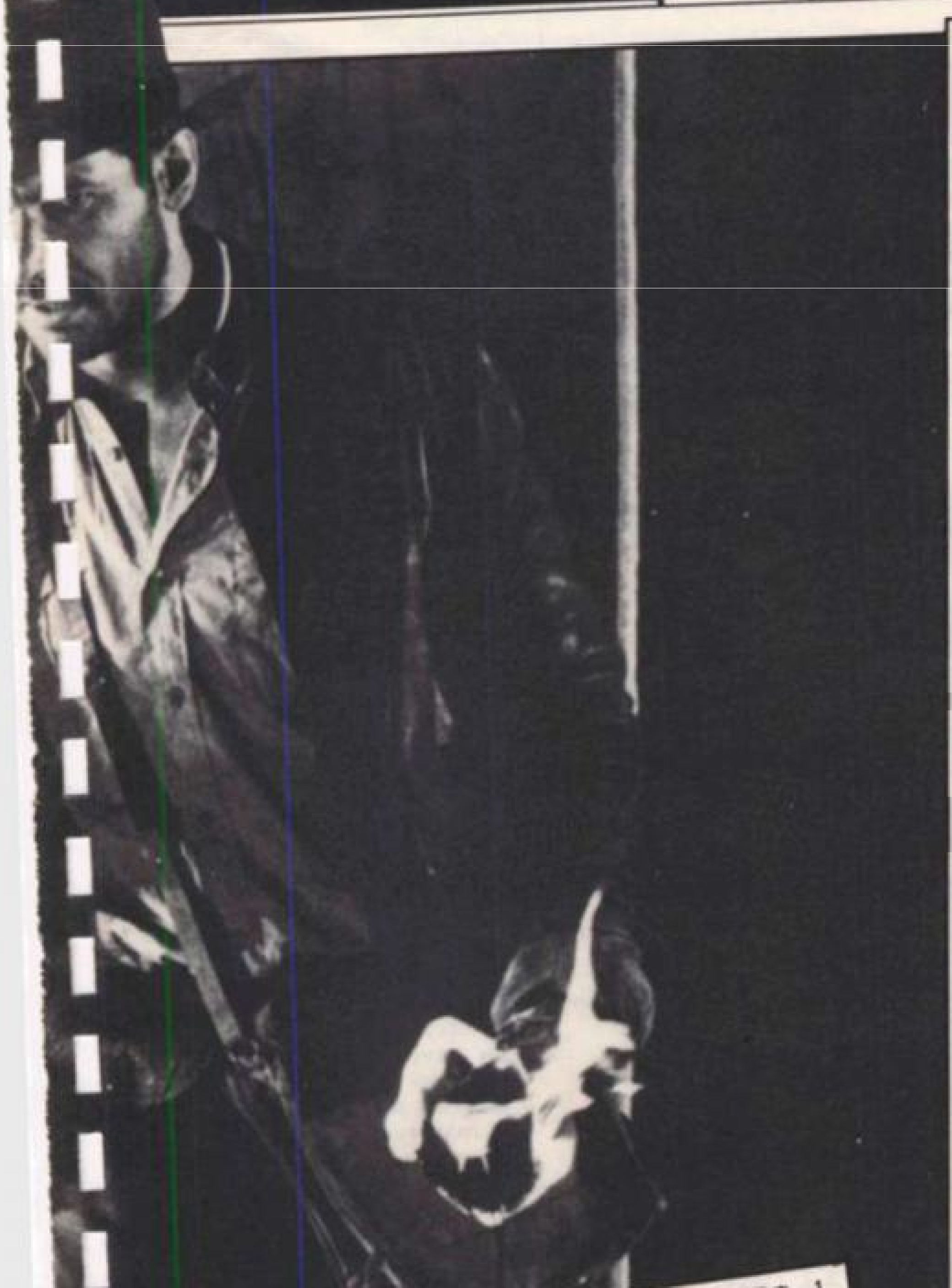
```
 20 REM **P.C. 9/82**
 30 CLS
 40 GOSUB 5000
 50 IF INKEY$<>"" THEN GOTO 50
 80 REM **MAIN GAME LOOP**
 90 LET N=1
100 LET Z=CODE INKEY$
110 IF Z>32 AND Z<37 THEN LET E
=Z-28
120 LET C=C+33*((E=6)-(E=7))+(E
=8)-(E=5)
130 IF PEEK C=23 THEN LET AB=AB
-4
140 IF AB<1 THEN LET AB=AB+50
150 IF PEEK C>=128 THEN GOTO 10
00
160 POKE C,136
170 POKE A(AB),14
180 LET A(AA)=C
190 LET AA=AA+1
200 LET AB=AB+1
210 IF AA>50 THEN LET AA=1
220 IF AB>50 THEN LET AB=1
300 REM **COMPUTERS MOVE**
400 IF PEEK (D+P(F))<24 THEN GO
TO 500
410 LET G=(2*INT (RND*2))-1
420 LET F=F+G
430 IF F>4 THEN LET F=F-4
440 IF F<1 THEN LET F=F+4
450 IF PEEK (D+P(F))<24 THEN GO
TO 500
```



```
46Ø LET F=F-(2*G)
47Ø IF F>4 THEN LET F=F-4
48Ø IF F<1 THEN LET F=F+4
49Ø IF PEEK (D+P(F))>23 THEN GO
TO 2ØØØ
5ØØ LET D=D+P(F)
51Ø POKE D,168
52Ø POKE B(BB),14
53Ø LET B(BB)=D
54Ø LET BB=BB+1
55Ø IF BB>1Ø THEN LET BB=1
56Ø REM **CHECK LENGTH**
6ØØ LET N=N+1
61Ø IF N<LEVEL THEN GOTO 1ØØ
62Ø POKE A(AB),14
63Ø LET AB=AB+1
64Ø IF AB>5Ø THEN LET AB=1
65Ø IF ABS (AA-AB)<1 THEN GOTO
1ØØØ
66Ø GOTO 9Ø
1ØØØ REM **COMPUTER WINS**
1Ø1Ø PRINT AT 23,Ø;"THE COMPUTER
 WINS"
1Ø2Ø POKE 16418,2
1Ø3Ø FOR M=1 TO 15Ø
1Ø4Ø NEXT M
1Ø5Ø RUN
2ØØØ REM **PLAYER WINS**
2Ø1Ø PRINT AT 23,Ø;"YOU WIN"
2Ø2Ø POKE 16418,2
2Ø3Ø FOR M=1 TO 15Ø
2Ø4Ø NEXT M
2Ø5Ø RUN
499Ø REM INSTRUCTIONS+SET VARS
5ØØØ PRINT TAB 5;"** SNAKE TRAP
**"
5Ø1Ø DIM A(5Ø)
5Ø2Ø DIM B(1Ø)
5Ø3Ø LET AA=1Ø
5Ø35 LET AB=1
5Ø8Ø LET P(2)=33
5Ø9Ø LET P(3)=-P(1)
51ØØ LET P(4)=-P(2)
511Ø LET DFILE=PEEK 16396+256*PE
EK 16397
512Ø FOR N=1 TO 5Ø
513Ø LET A(N)=16514
514Ø NEXT N
515Ø FOR N=1 TO 1Ø
516Ø LET B(N)=16514
517Ø LET E=8
518Ø LET F=1
516Ø LET B(N)=16514
517Ø LET E=8
518Ø LET F=1
519Ø LET C=DFILE+5+33*INT ((RND*
16)+3)+(INT (RND*2Ø))
52ØØ LET D=C+33
6ØØØ REM ****INSTRUCTIONS****
```



```
6Ø1Ø PRINT AT 2,Ø;"THE OBJECT OF
 THIS GAME"
6Ø2Ø PRINT "IS TO MOVE YOUR SNAK
E:"
6Ø3Ø PRINT "SHOWN: ";CHR$ 136;CH
R$ 136;CHR$ 136
6Ø4Ø PRINT
6Ø5Ø PRINT "TO TRY AND TRAP THE
COMPUTERS"
6Ø6Ø PRINT "SHOWN: ";CHR$ 168;CH
R$ 168;CHR$ 168;CHR$ 168
6Ø7Ø PRINT
6Ø8Ø PRINT "YOU CAN MOVE USING T
HE ARROWED"
6Ø9Ø PRINT "KEYS 5,6,7 AND 8"
61ØØ PRINT
611Ø PRINT "YOU WILL GROW IF YOU
 EAT THE"
612Ø PRINT "SQUARES MARKED *"
613Ø PRINT "BUT IF YOU FALL INTO
 THE"
```



```
6140 PRINT "SNAKE PITS (";CHR$ 1
28;") OR HIT THE WALL"
6150 PRINT "YOU WILL DIE"
6159 REM * CHARACTERS PRINTED  *
       * IN NEXT LINE ARE IN *
       * INVERSE.            *
6160 PRINT AT 21,7;"press a key"
6170 IF INKEY$="" THEN GOTO 6170
6180 CLS
6190 PRINT "BEWARE...."
6200 PRINT "IF YOU RUN INTO THE
OTHER SNAKE"
6210 PRINT "YOU WILL DIE AND IF
YOU RUN OUT"
6220 PRINT "OF FOOD (*) YOU MAY
STARVE."
6230 PRINT
6240 PRINT "WHAT LEVEL OF DIFFIC
ULTY"
6250 PRINT "WOULD YOU LIKE TO TR
Y???"
6260 PRINT "(1=EASY TO 9=IMPOSSI
BLE)"
6270 LET Z=CODE INKEY$-28
6280 IF Z<1 OR Z>9 THEN GOTO 627
0
6290 PRINT "LEVEL...";CHR$ (Z+15
6)
6300 PRINT
6309 REM * CHARACTERS PRINTED  *
       * IN LINES 6310/30/50/*
       * 70 AND 7010/50 ARE  *
       * IN INVERSE.         *
6310 PRINT TAB 4;"please wait wh
ile the"
6320 PRINT
6330 PRINT TAB 8;"screen is set"
6340 PRINT
6350 PRINT TAB 5;"this will only
 take"
6360 PRINT
6370 PRINT TAB 8;"a few seconds"
6380 FOR N=1 TO 100
6390 NEXT N
6400 FAST
6410 CLS
6420 LET LEVEL=13-Z
6990 REM ****SET UP BOARD****
7000 POKE 16418,0
7010 PRINT " snake trap
                "
7020 FOR N=1 TO 22
7030 PRINT CHR$ 128;":::::::::::
:::::::::::::::::::";CHR$ 128
7040 NEXT N
7050 PRINT "
                "
7060 FOR N=1 TO 20
7070 PRINT AT INT (RND*20)+1,INT
 (RND*30)+1;"*"
7080 NEXT N
7090 FOR N=1 TO 10
7100 PRINT AT INT (RND*20)+1,INT
 (RND*30)+1;" "
7110 NEXT N
7120 FOR N=1 TO 3
7130 POKE (C+N),14
7140 POKE (D+N),14
7150 NEXT N
7160 SLOW
7170 PRINT AT 0,16;"PRESS A KEY"
7174 REM * CHARACTERS PRINTED  *
       * IN NEXT 2 LINES ARE *
       * IN INVERSE.         *
7175 PRINT AT 0,16;"press a key"
7180 IF INKEY$="" THEN GOTO 7170
7190 PRINT AT 0,15;"good luck...
"
7200 RETURN
9998 SAVE "SNAKE TRAP"
9999 RUN
```

ZX designer

Make the most of your Spectrum's designing facilities courtesy of Chris Timson of East London.

This program allows you to create designs on your Spectrum using the PLOT, DRAW, and OVER commands.

After specifying the BORDER, PAPER and INK colours, you are invited, through user friendly instructions, to either plot a point, draw a line, or plot and draw over a line (in order to remove any mistakes). It is worth noting that DRAW always draws from the last point plotted, so that if you want a line separate from the last point drawn you must plot a point first.

On this last point, one of the major drawbacks (pun intended) of the DRAW command is that the co-ordinates of the end of the line are offsets of the last point plotted. However, it is quite easy to programme around this by using the two system variable, bytes 23677 (which holds the x co-ordinate of the last point plotted) and 23678 (which holds the y co-ordinate of the last point plotted).

The functions defined in lines 10 and 20 subtract those held in these bytes from the absolute screen co-ordinates you wish to draw to and gives the offset values required by the DRAW command. This method has ramifications for graphics work far beyond the scope of this relatively simple, if enjoyable, program and you should be able to make very good use of it in other graphics applications.




```
  10 DEF FN x(a)=a-PEEK 23677
  20 DEF FN y(a)=a-PEEK 23678
  30 BORDER 1: PAPER 1: INK 7: C
LS
  40 GO TO 3000
 100 INPUT "Enter 1 to plot, 2 t
o draw ";a
 110 IF a<1 OR a>2 THEN GO TO 10
0
 120 GO TO 1000*a
 999 REM **plot coords**
1000 INPUT "Enter 0 to plot, 1 t
o plot OVER ";a
1010 IF a<0 OR a>1 THEN GO TO 10
00
1020 INPUT "Enter plot coords ";
b;" ";c
1030 IF b<0 OR b>255 OR c<0 OR c
>175 THEN GO TO 1000
1040 PLOT OVER a;b,c
1050 GO TO 100
1999 REM **draw coords**
2000 INPUT "Enter 0 to draw, 1 t
o draw OVER ";a
2010 IF a<0 OR a>1 THEN GO TO 20
00
2020 INPUT "Enter draw coords ";
b;" ";c
2030 IF b<0 OR b>255 OR c<0 OR c
>175 THEN GO TO 2020
2040 DRAW OVER a;FN x(b),FN y(c)
2050 GO TO 100
2999 REM **instructions**
3000 FOR m=1 TO 11: FOR n=0 TO 1
3010 PRINT TAB RND*7; PAPER 2; I
NK 7; FLASH n;"ZX Designer"
3015 BEEP .025,m*2
3020 NEXT n: NEXT m
3030 PRINT AT 9,20;"Press any ";
AT 11,20;"letter to";AT 13,20;"c
ontinue"
3040 BEEP .025,0: PAUSE 3: IF IN
KEY$="" THEN GO TO 3040
3050 CLS
3060 PRINT AT 5,3;"This program
allows you to "
3070 PRINT "create designs using
PLOT, DRAW and OVER commands. Th
e DRAW command works in absolute
 screen coordinates."
3080 PRINT '"   Please enter BOR
DER colour   (0 to 7) ";
3090 INPUT a: IF a<0 OR a>7 THEN
 GO TO 3090
3100 PRINT a
3110 PRINT '"   Now enter PAPER
colour ";
3120 INPUT b: IF b<0 OR b>7 THEN
 GO TO 3120
3130 PRINT b
3140 PRINT '"   And INK colour "
;
3150 INPUT c: IF c<0 OR c>7 THEN
 GO TO 3150
3160 PRINT c
3170 PRINT '"   Further instruct
ions are     given in the INPUT
requests. Press any letter to co
ntinue."
3180 PAUSE 0
3190 BORDER a: PAPER b: INK c: C
LS
3200 GO TO 100
```

AGF

PRICE BREAKTHROUGH

EXCLUSIVE MAIL ORDER SPECIAL OFFER



9.95

AGF Joystick Interface II for Spectrum or ZX81

Now the AGF Interface II is even better value. Since we pioneered the cursor-key interface in October 1982 there are now over 100 games or utility programs with either the AGF option or cursor key controlled – that makes it unbeatable at this new low price.

Still incorporating the all important rear expansion connector which means other peripherals can be connected at the same time i.e Ram Packs, Printers, Speech Units etc, and of course the key replication principle used guarantees this will never conflict electrically with any other add-ons.

When you feel like upgrading to a more flexible joystick system you can claim 12.5% off the purchase price of our Programmable Interface.

All AGF Joystick interfaces will work with any industry standard joysticks using 9-pin 'D' plugs – including Quick-Shot II and the new Trackball controllers.

Don't settle for less .... choose AGF.

Joysticks

The choice is yours. The standard QuickShot is excellent value incorporating suction cups for sure-footed precision and a comfortably contoured handle with a convenient top firing button, plus conventional button in the base.

Four feet of cord terminates in the industry standard 9-pin 'D' plug for compatibility with our interfaces or direct connection to a selection of home micros.

The new QuickShot II has improved styling with a trigger type firing control as well as the top firing action and a broader base for greater suction stability. The rapid fire switch, which allows a continuous stream of shells where normally a button press is required for each one, is located in the base for easy access in use.

Cord length and termination is as the standard QuickShot controller.



8.95

QuickShot

QuickShot II



11.95

12.5% discount off our Programmable Interface~valid 1 year on orders over £18

PLUS free m/c arcade game with 1st 100 orders Spectrum only

so HURRY... post free both ways

Please allow up to 28 days from receipt of your order although we normally despatch from stock within 7 days. All AGF products are guaranteed for 12 months and are supplied on a 14 day money back undertaking. AGF Hardware, Bognor Regis, West Sussex. Telephone: (0243) 823337.

FROM: MR/MRS/MISS

ADDRESS

I ENCLOSE A CHEQUE/POSTAL ORDER PAYABLE TO AGF HARDWARE FOR £

SEND (NO STAMP NEEDED) TO: A.G.F. HARDWARE, DEPT. ZX

FREEPOST, BOGNOR REGIS, WEST SUSSEX, PO22 9BR

QTY	ITEM	ITEM PRICE	TOTAL
	AGF JOYSTICK INTERFACE II	9.95	
	QUICKSHOT JOYSTICK	8.95	
	QUICKSHOT II JOYSTICK	11.95	
ONE	DISCOUNT VOUCHER IF ORDER EXCEEDS £18	FREE	
	ZX81 ☐ ZX SPECTRUM ☐ Please tick	FINAL TOTAL	

Semaphore

We put the flags out for this fascinating program from Philip Jennings of Birmingham.

Did you know that people were transmitting and receiving long distance messages years before CB was invented! We've all seen old war films where semaphore signals are sent from ship to ship to relay vital information. Well, now you can learn the code and decipher those signals with the aid of this ZX81 program designed for the expanded machine.

All the relevant signals are assigned to S$ and the diagram shows what each signal should be and helps should you make a mistake. On running the program you are asked to select:

1 Type message	Must be selected to enter the message you want to send.
2 Send semaphore	Sends the message you have just input.
3 Test semaphore	Sends message one piece at a time. Letter must be input before it can proceed and mistakes are marked with an inverse E. The correct message is displayed on completion.

Note that the computer sends the authentic 'call up' before sending any message and each number is preceded by the 'number follows' signal. This is automatic. Incidentally, you can enter messages 'in secret' when they are not displayed on the screen before testing.

Entering the program

Type in Listing 1 first, exactly as printed. Line 1 *must* contain at least 152 characters. RUN the program and enter each of the numbers in Fig 1 one at a time, ie 4 newline 2 newline etc.

These numbers are *not* machine code but are the respective array element positions of each of the flags. It is best to check your work now, to be wrong would give erroneous information to the user. So, enter Listing 1a, deleting line 3 (Do not new) and RUN. Check the numbers displayed with those printed to make sure you have entered them correctly.

Now delete lines 2, 4 and 5 and enter the main program, Listing 2. Once you have RUN the program and checked with some test messages then SAVE it by typing GOTO 4000. On subsequent loadings the program will auto run and be instantly ready for use as it will not have to set up array S$ again. The last message entered will be saved as well.

```
1 REM .......................
..............................
..............................
..............................
..............................
.
2 FOR I=16514 TO 16665
3 INPUT A
4 POKE I,A
5 NEXT I
```

Listing 1

```
2 FOR I=16514 TO 16665
4 PRINT PEEK I;" ";
5 NEXT I
```

Listing 1a

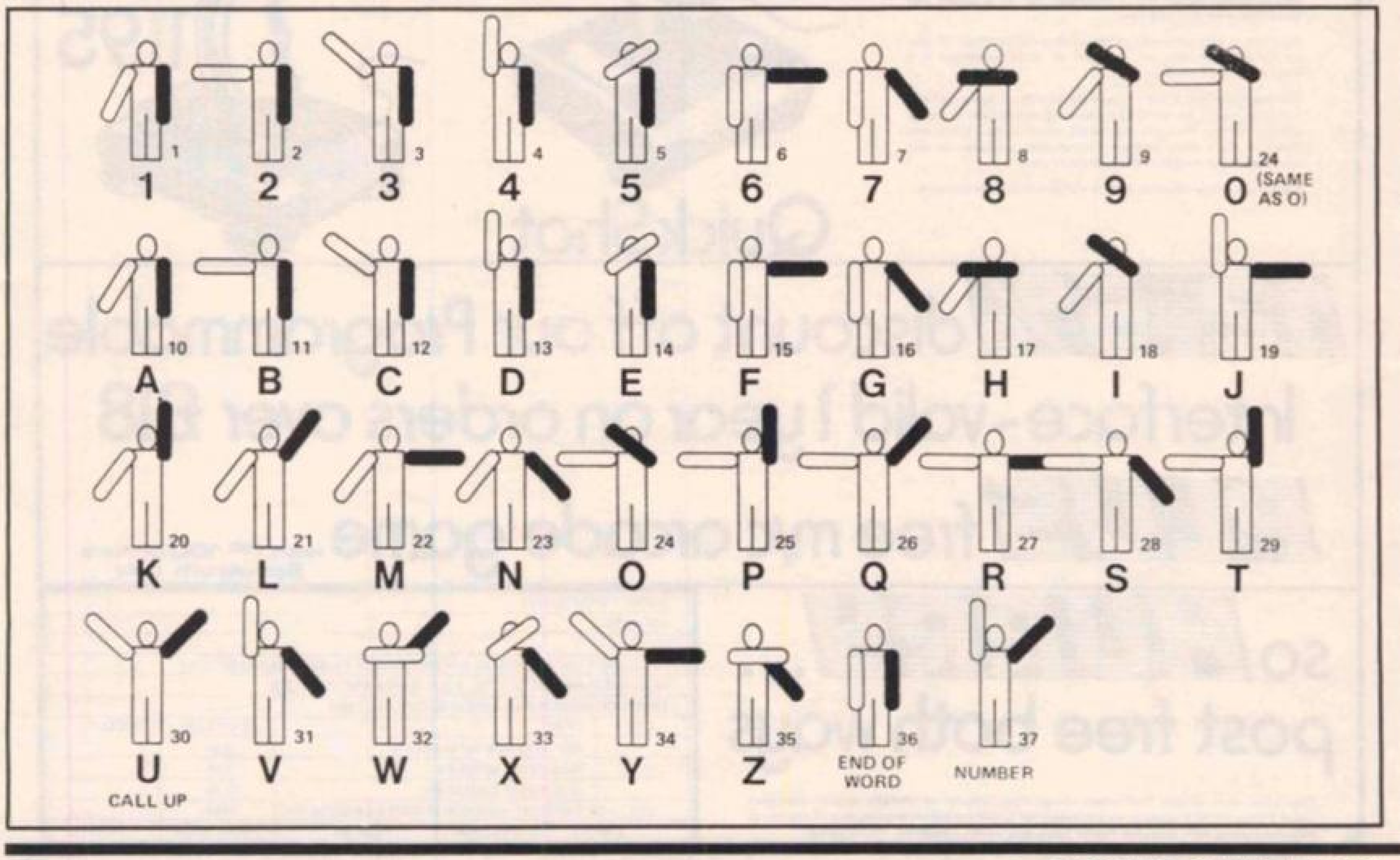


Fig 1

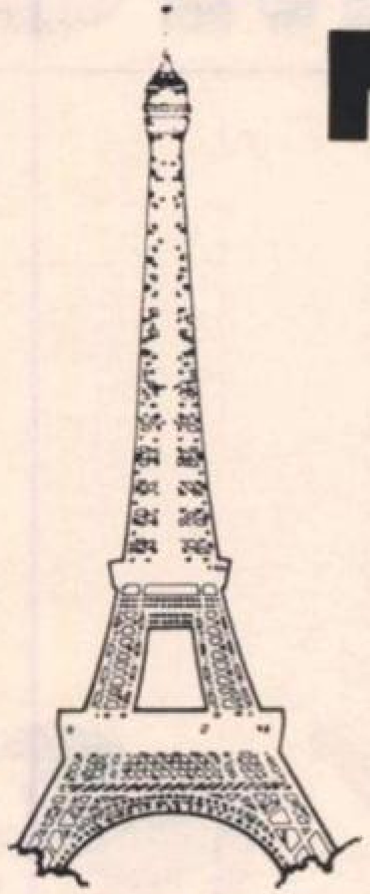
```
4 2 1 3 3 1 4 3 2 2 4 3 1 3 4 3
2 4 4 3 4 3 3 5 4 3 4 4 4 2 3 1
4 2 2 2 4 2 4 3 3 1 4 3 2 2 4 3
1 3 4 3 2 4 4 3 4 3 3 5 4 3 4 4
4 2 3 1 4 2 2 2 1 3 3 5 4 2 1 3
4 2 2 4 4 2 3 5 4 2 4 4 3 1 2 2
3 1 1 3 3 1 2 4 3 1 3 5 3 1 4 4
2 2 1 3 2 2 2 4 1 3 4 4 3 5 2 4
2 4 4 4 2 2 3 5 3 5 4 4 4 3 4 3
1 3 2 4 4 4 4 4
```

Listing 2

```
   5 FAST
  10 DIM S$(38,4,5)
  20 FOR I=1 TO 38
  30 LET S$(I,2,3)="O"
  40 LET S$(I,3,3)="A"
  50 LET S$(I,4,3)="I"
  60 NEXT I
  70 LET X=16514
  80 FOR I=1 TO 38
  90 LET S$(I,PEEK X,PEEK (X+1))
=CHR$ 8
 100 LET S$(I,PEEK (X+2),PEEK (X
+3))=CHR$ 128
 110 LET X=X+4
 120 NEXT I
 125 LET S$(36,4,3)=CHR$ 137
 130 LET S$(38,4,4)=" "
 140 SLOW
2000 CLS
2010 PRINT TAB 10;"SEMAPHORE";TA
B 10;"*********"
2030 PRINT
2040 PRINT "SELECT OPTION:"
2050 PRINT
2060 PRINT "1  TYPE MESSAGE"
2070 PRINT "2  SEND SEMAPHORE ME
SSAGE"
2080 PRINT "3  TEST SEMAPHORE"
2100 LET I$=INKEY$
2110 IF I$<"1" OR I$>"3" THEN GO
TO 2100
2120 IF I$<>"1" THEN GOTO 2500
2300 CLS
2310 PRINT "SELECT OPTION"
2320 PRINT
2330 PRINT "1  TYPE MESSAGE IN S
ECRET"
2340 PRINT "2  TYPE MESSAGE"
2350 LET I$=INKEY$
2360 IF I$<>"1" AND I$<>"2" THEN
 GOTO 2350
2380 CLS
2385 LET N$=""
2390 PRINT "ENTER MESSAGE, SEPAR
ATE WORDS   BY "".""  ,PRESS "","
" TO END."
2400 IF INKEY$<>"" THEN GOTO 241
0
2405 LET M$=INKEY$
2410 IF M$="" THEN GOTO 2405
2420 IF M$="," THEN GOTO 2000
2430 IF CODE M$=118 THEN LET N$=
N$( TO LEN N$-1)
2435 IF CODE M$=118 THEN GOTO 24
45
2440 LET N$=N$+M$
2445 IF I$="2" THEN PRINT AT 5,0
;N$;" "
2450 GOTO 2400
2500 CLS
2505 LET L$=""
2510 FOR I=1 TO 3
2520 LET P=30
2530 GOSUB 3000
2540 LET P=36
2550 GOSUB 3000
2560 NEXT I
2570 FOR I=1 TO LEN N$
2580 IF N$(I)="." THEN LET P=36
2590 IF N$(I)="." THEN GOTO 2660
2600 IF N$(I)<CHR$ 38 THEN GOSUB
 3040
2610 LET P=CODE N$(I)-28
2615 IF N$(I)="0" THEN LET P=24
2620 GOSUB 3000
2630 FOR A=1 TO 5
2640 NEXT A
2650 LET P=38
2660 GOSUB 3000
2670 IF I$="2" THEN NEXT I
2675 IF I$="2" THEN GOTO 2740
2680 LET M$=INKEY$
2690 IF M$="" THEN GOTO 2680
2700 LET L$=L$+M$
2710 PRINT AT 20,1;L$
2720 IF L$(I)<>N$(I) THEN PRINT
AT 21,I;"e"
2730 NEXT I
2732 FOR I=1 TO 35
2733 NEXT I
2735 PRINT AT 18,1;N$
2740 LET P=36
2742 GOSUB 3000
2745 FOR I=1 TO 50
2750 NEXT I
2990 GOTO 2000
3000 FOR J=1 TO 4
3010 PRINT AT 10+J,12;S$(P,J)
3020 NEXT J
3030 RETURN
3040 LET P=37
3050 GOTO 3000
4000 SAVE "SEMAPHORE"
4010 GOTO 2000
```

Parlez-vous Micro-PROLOG?

Our reviewer has taken a look at an up and coming language from Sinclair.



Although I enjoy learning and exploring new computer languages, when I saw the Sinclair Micro-PROLOG package I was a little worried to say the least at the size of the manual and its complexity.

Still thinking that I had let myself in for a pretty stiff little number, I dived into the introductory manual which is much thinner and aimed toward your average Specy user. After three pages of how to switch your computer on and how to load a program, I finally found out about the SIMPLE language. SIMPLE is a program which changes the way you use Micro-PROLOG so that it is not too confusing at first. I was quite impressed, I managed to follow what the manual was telling me without becoming hopelessly lost.

PROLOG itself is a logically-structured language; that is, it is no where near as messy as BASIC, but then it is not as easy to grasp as BASIC. Consider the Micro-PROLOG line:

which (x : x larger London)

Taking this line step by step, first we have the command 'which'. This is asking Micro-PROLOG a question. The question is contained within the brackets and is in the form 'output format:argument'. In the example above I am dealing with cities and am looking for all the cities 'larger' than London. The word 'larger' in itself doesn't mean anything to Micro-PROLOG, what it is doing is just looking through its database and finding any places where a city has been specified as larger than London.

The large green manual which accompanied the cassette was far beyond me. I found I could use the program quite well enough just by following the small booklet.

The main use of Micro-PROLOG is obviously 'do-it-yourself' database. Tailoring the filing system to your own requirements is very simple. Once you have a knowledge of Micro-PROLOG, you can create a database far better than any commercially available — because you know exactly what you want from the system. Table 1 shows the commands available and their syntax.

The loading and saving operations are similar to that of the BBC in that they load in blocks and it is very slow. It is quite reliable though, and the programs you write are not likely to be too large anyway.

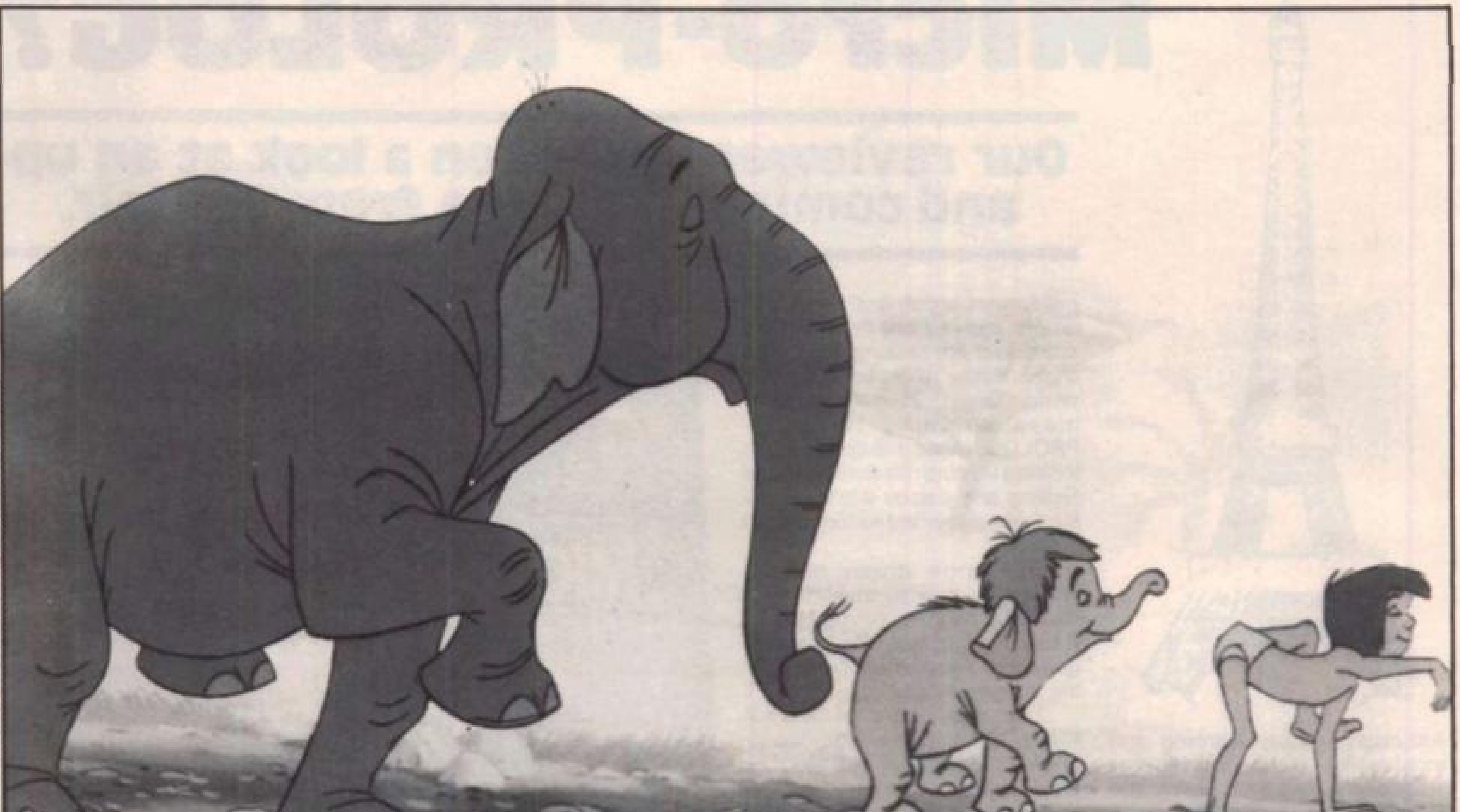
Micro-PROLOG was written by Logic Programming Associates Ltd, and is available from most high street stores under the Sinclair label priced at £24.95.



Table 1. Commands and Syntax

add <u>(argument)</u>	
list <u>description</u>	
is <u>(argument)</u>	This will produce a YES or NO response
which (output format: argument)	This will produce all objects, etc, which fit the argument.
delete <u>description</u>	This takes out single arguments
kill <u>description</u>	This takes out whole blocks
save <u>program name</u>	
load <u>program name</u>	

Follow Me



Photograph courtesy of Walt Disney Productions Ltd.

A game calculated to give you brain strain from Stephen Proctor in Avon.

This is an unusual version of the old electronic game in that PLOT and DRAW are used to create a very large solid circle. A loop is set up for the X co-ordinate then a line is drawn using two numbers stored as DATA. These numbers are the Y co-ordinate for the start of the line and the length of the line.

Another unusual feature is that up to eight players can play! Each player has a score which is equal to the number of colours in his sequence, provided that he gets it right of course. The game is simple but addictive; all you have to do is watch the sequence of colours presented by the computer and then repeat them in the same order.

Full instructions are included in the program and when prompted with "Up to" at the start of each game enter the score which one of the players must reach before the game ends.

```
  10 GO SUB 2000: GO SUB 3000
  20 FOR q=1 TO 162: READ x,y: P
LOT q+47,x: DRAW 0,y: NEXT q
  30 FOR p=1 TO np: PAUSE 150: P
RINT AT 0,0;"Your go, ";n$(p)
  35 DIM w(l(p))
  40 FOR c=1 TO l(p)
  50 LET qz=INT (RND*4)+1
  60 LET col=16*(qz=1 OR qz=3)
  70 FOR l=0 TO 10: IF qz>2 THEN
 FOR l=11 TO 21
  80 PRINT AT l,col; OVER 1; INK
 VAL "1246"(qz);"
": NEXT l
  90 BEEP .7,qz
 100 LET w(c)=qz
 110 FOR l=0 TO 10: IF qz>2 THEN
 FOR l=11 TO 21
 120 PRINT  OVER 1; INK 0;AT l,c
ol;"                ": NEXT l
 130 NEXT c
 140 PAUSE 100: FOR c=1 TO l(p)
 150 LET a$=INKEY$: IF a$<>"r" A
ND a$<>"g" AND a$<>"b" AND a$<>"
y" THEN  GO TO 150
 160 IF a$<>"brgy"(w(c)) THEN  G
O TO 300
 165 PRINT AT 0,0;"
                         ";AT 0,0;"Cor
```

```
rect!!!                  "
 167 BEEP .7,w(c)
 168 IF INKEY$<>"" THEN  GO TO 1
68
 170 NEXT c
 180 LET s(p)=s(p)+l(p): LET l(p
)=l(p)+1
 185 FOR d=1 TO np: PRINT AT 21,
(d-1)*4;s(d): NEXT d: FOR d=1 TO
 np: IF s(d)>=up THEN  GO TO 500
 187 NEXT d
 190 NEXT p
 210 GO TO 30
 300 LET l(p)=l(p)-1: GO TO 185
 500 PAPER 0: BORDER 0: INK 7: C
LS : PRINT AT 11,0; FLASH 1;"
   You've won, ";n$(d);"      "
 510 RESTORE 9030: FOR q=1 TO 12
: READ x,y: BEEP x/8,y: BORDER R
ND*7: NEXT q
 515 PRINT AT 15,0;"Scores:"''
 517 FOR x=1 TO np: PRINT AT 17,
(x-1)*4; FLASH (x=d);s(x)
 520 NEXT x: BORDER 0: PAUSE 0:
RUN
 999 REM **********************
         * Solid circle Data   *
         **********************
1000 REM
1030 DATA 89,-4,77,20,72,30,67,4
0,63,48,61,52,58,58,56,62,53,68
1040 DATA 51,72,49,76,47,80,46,8
2,44,86,43,88,42,90,40,94,39,96
1050 DATA 37,100,36,102,35,104,3
4,106,33,108,32,110,31,112,30,11
4,29,116
1060 DATA 28,118,27,120,26,122,2
5,124,24,126,24,126,23,128,22,13
0,22,130
1070 DATA 21,132,20,134,19,136,1
9,136,18,138,17,140,17,140,16,14
2,16,142
1080 DATA 15,144,15,144,14,146,1
4,146,13,148,13,148,13,148,12,15
0,12,150
1090 DATA 11,152,11,152,10,154,1
0,154,10,154,10,154,9,156,9,156,
9,156
1100 DATA 9,156,9,156,8,157,8,15
8,8,158,8,158,8,158,8,158,7,160
1110 DATA 7,160,7,160,7,160,7,16
0,7,160,7,160,7,160,7,160,7,160
1120 DATA 7,160,7,160,7,160,7,16
0,7,160,7,160,7,160,7,160,7,160
1130 DATA 7,160,8,158,8,158,8,15
8,8,158,8,158,9,157,9,156,9,156
1140 DATA 9,156,9,156,9,156,10,1
54,10,154,10,154,10,154,11,152,1
1,152
1150 DATA 12,150,12,150,13,148,1
3,148,13,148,14,146,14,146,15,14
4,15,144
1160 DATA 16,142,16,142,17,140,1
7,140,18,138,19,136,19,136,20,13
4,21,132
1170 DATA 22,130,22,130,23,128,2
4,126,24,126,25,124,26,122,27,12
0,28,118
1180 DATA 29,116,30,114,31,112,3
2,110,33,108,34,106,35,104,36,10
2,37,100
1190 DATA 39,96,40,94,42,90,43,8
8,44,86,46,82,47,80,49,76,51,72
1200 DATA 53,68,56,62,58,58,61,5
2,63,48,67,40,72,30,77,20,89,-4
2030 INK 0: PAPER 7: CLS : FOR x
=0 TO 5: BORDER x: PRINT AT 1,x+
12; INK x;"PSYMON"(x+1): BEEP .1
,x: NEXT x
2040 PAUSE 50: FOR x=96 TO 143:
PLOT x,159: BORDER INT (RND*8):
BEEP .05,x-110: NEXT x
2050 BORDER 7
2060 PRINT AT 3,0;"I will flash
an ever-increasing series of col
oured lights accom-pianied by a
different note for each colour.T
he players have to remember the
sequence."
2070 PRINT '"  To enter a colour
, press the  initial letter of t
hat colour,  eg"'' INK 1;"(b)lue
  "; INK 2;"(r)ed   "; INK 6;"(y)
ellow  "; INK 4;"(g)reen"
2075 PAUSE 1000
2080 PRINT AT 19,0; FLASH 1; PAP
ER 5;"   Press any key to contin
ue    "
2090 FOR b=0 TO 7: BEEP .05,b: I
F INKEY$="" THEN  NEXT b: GO TO
2090
2100 CLS : RETURN
3000 BORDER 5: PRINT AT 11,4;"Ho
w many players (1-8)?"
3010 LET a$=INKEY$: IF a$<"1" OR
 a$>"8" THEN  GO TO 3010
3015 PRINT AT 0,0;"
                               "

3020 LET np=VAL a$: DIM l(np): D
IM n$(np,10): DIM s(np)
3030 FOR x=1 TO np: INPUT ('"Nam
e of player ";x;);n$(x): LET l(x
)=4: NEXT x
3040 INPUT '''Up to ";up
3050 GO TO 2080
9030 DATA 1,5,1,2,1.5,-2,1.5,-2,
.75,-2,.75,0,.75,2,.75,4,1.5,5.1
,1.5,5,1.5,5,.75,2
```



	CM64	BBC	ELECTRON	DRAGON	SPECTRUM
CHUCKIE EGG	£7.90	£7.90	£7.90	£7.90	£6.90
CYLON ATTACK		£7.90	£7.90		
JUNGLE FEVER					£6.90
PHARAOHS TOMB		£7.90	£7.90		

Sold subiect to A & F Software Ltd. Full terms and conditions available on request

A&F Software

AVAILABLE FROM ALL GOOD COMPUTER SHOPS

John Menzies

Boots Selected Stores

Unit 8, Canalside Industrial Estate,
Woodbine Street East, Rochdale, Lancs.
OL16 5LB. Tel: 0706 341111

Quality programs from
HARTLAND SOFTWARE

Hartland Software's Famous
FOOTBALL POOLS PROGRAM
This program has proved exceptionally popular so far, winning several dividends! The program lists out, in order of preference, the sixteen most likely score draws; also the sixteen most likely home wins, draws (including 0-0) and away wins. The program will be ilitialised to the English and Scottish league tables, but you can use it on the Australian pools just as easily, by altering the built-in team names and saving them on another cassette. At time of press, this is the ONLY pools forecasting system that uses purely statistical methods!
FOR ZX81 (16K) or ZX SPECTRUM (48K) **£6.95**

MISTERMIND
Hartland Software's version of the popular board game, commonly known as "Bulls and Cows". Not only can you *play* the game with this program, but you can take the computer on. You will find it very hard to beat!
FOR ZX SPECTRUM (48K) **£6.95**

TOP DOG (formerly marketed by BOSOFT)
A game simulating the world of dog showing and breeding. You start with £500 to buy your first pair; the object of the game is to win the Cruft's Supreme Champion prize without going bankrupt!
FOR ZX SPECTRUM (48K) **£5.95**

TRAVELLING SALESMAN PROBLEM
Not a game, but a fascinating insight into the world of network theory. The travelling salesman problem is an old one. You have a number of towns (with their grid references); the object is to find the shortest route so that the salesman visits each town once, then returns to his original starting place. This program uses the 2-optimal method to tackle the problem.
FOR ZX SPECTRUM (48K) **£8.95**

HARTLAND SOFTWARE, (DEPT Z), 32 IVOR PLACE, LONDON NW1 6DA

AUF WIEDERSEHEN Chérie

SUPER GERMAN VOCABULARY TEST
& SUPER FRENCH VOCABULARY TEST

Tutorial Software have greatly improved their successful German & French language tutors.

These Programs provide an extensive basic vocabulary. By using the 'Add New Words' facility the range of vocabulary that can be learned becomes limitless.

If you are studying for G.C.E. 'O' level or just interested in languages let these programs take the pain out of learning.

Programs £6.95 each and are only available for the 48K Spectrum.

Features:-

- 550 Foreign Words.
- Nouns, Verbs, Adjectives & Adverbs.
- Gender & Accents.
- Learn Mode & Test Mode.
- Add New Words Facility.
- Score Record And Average Score.
- Words Checked By Language Teachers.

Cheques or P.O.'s to:-
Tutorial Software
P.O. Box 43,
Enquiry Office
22 - 29 Mill Street,
Newport,
Gwent,
NPT 3XZ.

University Software

LIBRARY OF ADVANCED MATH/STAT/ECON FOR SINCLAIR ZX81 AND SPECTRUM

TAPE 1 : MATRIX OPERATIONS (*), (+) **£9.95**
SIDE A : Inversion, multiplication, addition, subtraction, scalar multiplication and determinants of matrices and vectors within one single program. Any output can in turn be used as the input of the next operation without re-typing. Capacities : 16K ZX81 : 25x25, 16K Spectrum : 15x15, 48K Spectrum : 48x48.
Side B : Determinants of square matrices.

TAPE 2 : POLYNOMIALS (+) **£6.95**
SIDE A : Includes quadratic equations (as degree 2 polynomials) and Newton-Raphson and half-interval search methods for higher degree polynomials. Computes the real roots with 8 digits of precision. SIDE B : Plot of polynomials in any interval, values of real roots, extremum points.

TAPE 3 : INTEGRATION (+) **£6.95**
SIDE A : Simultaneous integration of two functions by Simpson's and trapezoidal rules. Also computes the area enclosed between the two functions. SIDE B : Plot of two functions in any interval. Integrated area shaded.

TAPE 4 : SUPER-REGRESSION (*) **£14.95**
SIDE A : A highly developed multivariate regression program with log/ ln option on each variable, allowing for exponential and geometric regressions. Displays estimated coefficients, standard errors, t-statistics, R^2, corrected R^2, F-statistic, degrees of freedom, Durbin-Watson statistic, variance-covariance matrix, matrix of correlation coefficients, INTERPOLATION and PLOT of residuals. Capacity examples (var.xobs.) : 16K ZX81 : 2x400, 5x200, 10x100, 16K Spectrum : 2x75, 5x30, 48K Spectrum : 2x1650, 5x900, 10x500. SIDE B : Plot of bivariate regressions, slope, intercept, R^2, standard deviation.

TAPE 5 : PROFESSIONAL LINEAR PROGRAMMING (*) **£14.95**
SIDE A : A user friendly optimisation program capable of handling all sorts of linear programming problems (any combination of $<=>$ constraints and $X_i>0$, $X_i<0$, $-\alpha< X_i <\alpha$ sign constraints). Displays the cannonical equivalent of the primal, values of slack variables, dual problem and its solution. Capacity examples (var.xcons.) : 16K ZX81 : 10x21, 15x18, 20x13, 16K Spectrum : 8x8, 48K Spectrum : 10x45, 25x35, 50x25. SIDE B : Solutions of simultaneous equations.

TAPE 6 : STATISTICS (*), (+) **£14.95**
SIDE A : STATISTICAL TESTS AND DISTRIBUTIONS. Chi-square, Students t-, F-, Mann-Whitney U, Wilcoxon's Signed Rank tests and Chi-square, Students t-, F-, Binomial, Poisson and Normal distributions. Data on many samples can be entered at once, sorted and basic statistics displayed. Capacity examples (samp.xobs.) : 16K ZX81 : 1x230, 2x110, 5x45, 16K Spectrum : not available, 48K Spectrum : 2x1300, 10x250, 20x125.
SIDE B : DESCRIPTIVE STATISTICS. Analysis of ungrouped or grouped data. Ungrouped data sorted and grouped. Output includes mean, mean deviation, standard error, skewness, kurtosis, moments, Pearson's coefficient, etc. A comprehensive histogram with up to 30 class intervals, class frequencies, midpoint values. Capacities : ZX81 : 450, 16K Spectrum : 200, 48K Spectrum : 2000.

TAPE 7 : BIBLIOFILE **£9.95**
An academic indexing program to keep record of books and articles. Loading and saving of files, printing and listing of up to 150 records in a file, searching for an entry and sorting records into alphabetical order. It is possible to modify, delete and add a record.

EDUCATIONAL SERIES

INTRODUCTION TO ECONOMICS SET (48K Spectrum only) : Basic concepts are introduced by extensive use of graphics, examples and excercises.

MICROECONOMICS I (+) **£9.95**
SIDE A : Demand and supply analysis, demand theory, cost-benefit analysis. SIDE B : Budget line analysis, indifference curves, decision making in a market economy.

MICROECONOMICS II (+) **£9.95**
SIDE A : Cobb-Douglas production function, linear programming and excercises on linear programming. SIDE B : Theory of the firm, elasticity of demand, CES production function.

MACROECONOMICS (+) **£9.95**
IS/LM curves, national income determination, aggregate demand and supply analysis, multipliers.

ECONOMIC GEOGRAPHY (+) **£9.95**
SIDE A : Von Thunen's model of agricultural land use. Locational rent and principle of diminishing returns. SIDE B : Models of agricultural and urban land use.

(*) Programs with INPUT, CHANGE, LPRINT and SAVE-DATA facilities. Spectrum versions of tapes 4 and 6 also embody MERGE/READ DATA facilities. A common data file can be used with these programs.
(+) Recommended for "A" Level, first year university and poly students.

PLEASE SEND A5 SIZE S.A.E. FOR INFORMATION ON MICRODRIVE-BASED PACKAGES.
Tapes 1-6 : £60. Introduction to economics set : £35.
All inclusive prices for the U.K. For overseas orders add 50p per cassette.
Dealers enquiries welcome.
Cheques payable to : **UNIVERSITY SOFTWARE, 29 St. Peter's Street, London N1 8JP. Telephone : 01-359 0978**

Youth club

Tired of fighting intergalactic troublemakers? Let Simon Springett put you in charge of the human type with this unusual simulation game.



Remember the Youth Club? You know, that place you used to go before you had a computer. Running one isn't as easy as you might think — you are responsible for recruitment, training, fundraising and balancing the books. If you fail at any of these tasks you lose the game. Just to make it more difficult, troublemakers will turn up from time to time and attempt to destroy all your hard work by causing damage and deterring new members.

The program is designed to be user-friendly, requiring single key operation without the need to ENTER data.

On the right lines

Here is a rundown of some of the more important lines in the program:

- 5 — Initial display and setting of variables to initial values.
- 500 — Information. Note the use of a FOR-NEXT loop with RESTORE and DATA to produce a neatly set out display without too many PRINT commands.
- 1000 — Displays current situation. With so many variables needing printing lots of PRINT commands is the easy option this time!
- 1500 — Runs the group for one month. Calculates new members, expenses etc.
- 2000 — Runs a fight. BEEP commands produce a siren effect and damages increase until B is pressed. The program then goes to 1500.
- 3000 — This section allows you to change the balance of the programme (ie the club programme!) to raise more funds, train new leaders etc.
- 4000 — Decides if there will be a fight this month. Fights occur at least one month in ten, more often if you have only just enough leaders.
- 5000 — Displays the fact that you've lost — ran out of money.
- 5500 — Displays the fact that you've lost — didn't get the required number of members (100) in the allowed time (30 months).
- 6000 — Displays the fact that you've won and offers a new game.
- 9000 — A subroutine that stops 'overrun' when a key is held down. It is used eg. to prevent you running two weeks when you meant to run only one. Release the key and press it again to run another week.

```
        YOUTH CLUB (Month 0)

0 Members           LAST MONTH:
1 Leaders           0 New members
'10 in bank         0 Members left
                    '0 was spent
                    '0 damages
                    '0 was raised

PROGRAMME:
Fund raising 0 hrs/mth
Social events 16 hrs/mth
Training 0 hrs/mth

PRESS
1 to run group for 1 month
2 to change programme
3 for info
4 to abandon this game

      Press a key to continue
```

```
            YOUTH CLUB

    Your Target: 100 members

      Time limit: 30 months

   Social events recruit members

     Raise funds or go broke

    Each member costs £1/mth

        Stop fights quickly

          Train leaders:

 10 man hours trains a leader

      Max 12 members/ leader

      Press a key to continue
```

```
   5 REM INITIAL DISPLAY
  10 CLS : PRINT  AT 3,11;"YOUTH CLUB"
  20 PRINT  AT 8,7;"@ S.Springett 1983"
  30 GO SUB 9000
 100 LET a=0: LET m=0: LET l=13: LET b=10
 110 LET f=0: LET s=16: LET t=0
 120 LET d=0: LET g=0: LET c=0: LET r=0: LET p=0
 500 CLS : REM INFORMATION
 505 RESTORE
 510 FOR n=0 TO 9
 520 READ a$
 530 PRINT  AT n*2,(32- LEN a$)/2;a$
 540 NEXT n
 550 DATA "YOUTH CLUB","Your Target: 100 members","Time limit: 30 months","Socia
l events recruit members","Raise funds or go broke","Each member costs £1/mth","
Stop fights quickly","Train leaders:","10 man hours trains a leader","Max 12 mem
bers/ leader"
 560 GO SUB 9000
1000 CLS : REM DISPLAY CURRENT SIT
1010 PRINT  AT 0,10;"YOUTH CLUB"
1020 PRINT  AT 0,21;"(Month ";a;")"
1030 PRINT  AT 3,0;m;" Members","LAST MONTH:"
1035 LET e= INT (l/10)
1040 PRINT  AT 4,0;e;" Leaders",r;" New members"
1050 PRINT  AT 5,0;"£";b;" in bank",p;" Members left"
1060 PRINT  AT 6,0,"£";c;" was spent"
1070 PRINT  AT 7,0,"£";d;" damages"
1071 PRINT  AT 8,0,"£";g;" was raised"
1075 PRINT  AT 9,0;"PROGRAMME:"
1080 PRINT  AT 10,0;"Fund raising ";f;" hrs/mth"
1090 PRINT  AT 11,0;"Social events ";s;" hrs/mth"
1100 PRINT  AT 12,0;"Training ";t;" hrs/mth"
1105 PRINT  AT 14,0;"PRESS"
1110 PRINT  AT 15,0;"1 to run group for 1 month"'"2 to change programme"'"3 for
info"'"4 to abandon this game"
1120 GO SUB 9000
1130 IF b<0 THEN  GO TO 5000
1140 IF a=30 AND m<100 THEN  GO TO 5500
1150 IF m>99 THEN  GO TO 6000
1160 IF  INKEY$ ="1" THEN  GO TO 4000
1170 IF  INKEY$ ="2" THEN  GO TO 3000
1180 IF  INKEY$ ="3" THEN  GO TO 500
1190 IF  INKEY$ ="4" THEN  CLS : PRINT  AT 5,7; FLASH 1;"Abandon this game?": PR
INT  AT 10,10;"C to confirm": PRINT  AT 12,3;"Any other key to continue": GO SUB
 9000: IF  INKEY$ ="c" THEN  RUN
1200 GO TO 1000
1490 REM RUN 1 MONTH
1500 LET r= INT (s*s* RND /16)
1501 IF r+m>e*12 THEN  LET r=(e*12)-m
1505 LET a=a+1
1510 LET p= INT (m*(16-s)* RND /64)
1520 LET g= INT (f* RND *m/2)
1530 LET c=m
1540 LET b=(b+g)-(c+d)
1550 LET l=l+(e*t)
1560 LET m=m+r-p
1700 GO TO 1000
2000 CLS : REM FIGHT
2010 PRINT  AT 0,13; FLASH 1;"FIGHT!"
2015 PRINT  AT 7,5;"Press B to break it up"
```



```
2016 PRINT  AT 4,9;"Damage now: £";d
2020 LET d=d+ INT ( RND *15)
2025 BEEP .4,24
2030 PRINT  AT 4,22;d
2050 BEEP .4,22
2060 IF  INKEY$  <> "b" THEN  GO TO 2020
2070 GO SUB 9000
2080 GO TO 1500
2990 REM CHANGE PROGRAMME
3000 CLS : PRINT  AT 0,8;"CHANGE PROGRAMME"
3010 PRINT  AT 2,1;"You have a total of 16 hrs/mth"
3020 PRINT  AT 6,0;"Fund raising: "'"Social events: ";s;'"Leadership Trg: ";t
3030 PRINT  AT 12,0;"1 to increase"'"2 to decrease"'"3 to enter new value"
3035 PRINT  AT 6,14; FLASH 1;f;" "
3040 GO SUB 9000
3045 IF  INKEY$ ="3" THEN  GO TO 3100
3050 IF  INKEY$ ="1" THEN  LET f=f+1
3051 IF  INKEY$ ="2" THEN  LET f=f-1
3055 IF f=17 THEN  LET f=16
3056 IF f=-1 THEN  LET f=0
3060 GO TO 3035
3100 PRINT  AT 6,14;f;"  "
3110 PRINT  AT 7,15; FLASH 1;s;" "
3120 GO SUB 9000
3125 IF  INKEY$ ="3" THEN  GO TO 3200
3130 IF  INKEY$ ="1" THEN  LET s=s+1
3135 IF  INKEY$ ="2" THEN  LET s=s-1
3140 IF s=-1 THEN  LET s=0
3150 IF s=17 THEN  LET s=16
3160 GO TO 3110
3200 PRINT  AT 7,15;s;"  "
3210 IF s+f>16 THEN  CLS : PRINT  AT 8,1;"Your maths is a little rusty!": PAUSE
100: GO TO 3000
3220 LET t=16-(s+f)
3230 PRINT  AT 8,16;t
3240 PRINT  AT 18,10;"C to confirm": PRINT  AT 19,4;"Any other key to change"
3250 GO SUB 9000
3260 IF  INKEY$ ="c" THEN  GO TO 1000
3270 GO TO 3000
4000 LET d=0: REM FIGHT?
4010 FOR n=1 TO 1+ INT (m/(e*12))
4020 IF  RND <.1 AND m>1 THEN  GO TO 2000
4030 NEXT n
4040 GO TO 1500
5000 CLS : PRINT  AT 3,11; FLASH 1;"BANKRUPT!"
5010 PRINT  AT 5,9;"YOU'RE BROKE!": PRINT  AT 6,6;"You owe "";-b;" pounds"
5020 GO TO 6010
5500 CLS : PRINT  AT 3,11; FLASH 1;"YOU LOST!"
5510 PRINT  AT 5,8;"Only ";m;" Members": PRINT  AT 6,10;"in 30 months"
5520 GO TO 6010
6000 CLS : PRINT  AT 3,12; FLASH 1;"YOU WON!"
6005 PRINT  AT 5,10;m;" members": PRINT  AT 6,10;"in ";a;" months"
6010 PRINT  AT 8,2;"Press any key to play again"
6020 GO SUB 9000
6030 RUN
9000 IF  INKEY$  <> "" THEN  GO TO 9000
9010 PRINT  AT 21,4;"Press a key to continue"
9020 IF  INKEY$ ="" THEN  GO TO 9020
9030 RETURN
```





Bar Charts Revisited

Don't just store that information inside your ZX81, display it with the aid of Norman Brooks in Kent.

When confronted with a mass of numeric data it is often easier to understand and analyse if it is drawn as a bar chart. This program enables you to chart up to twenty-four numeric quantities and can handle numbers in the range of 0 to 9999. You can manipulate the vertical scale at will to suit the commodity you are charting.

Menu driven, it enables you to set up a bar chart and, once created, add and change data quickly and easily. You might for example use it to:

Chart annual income.
Graph media statistics (eg Retail Price Index.)

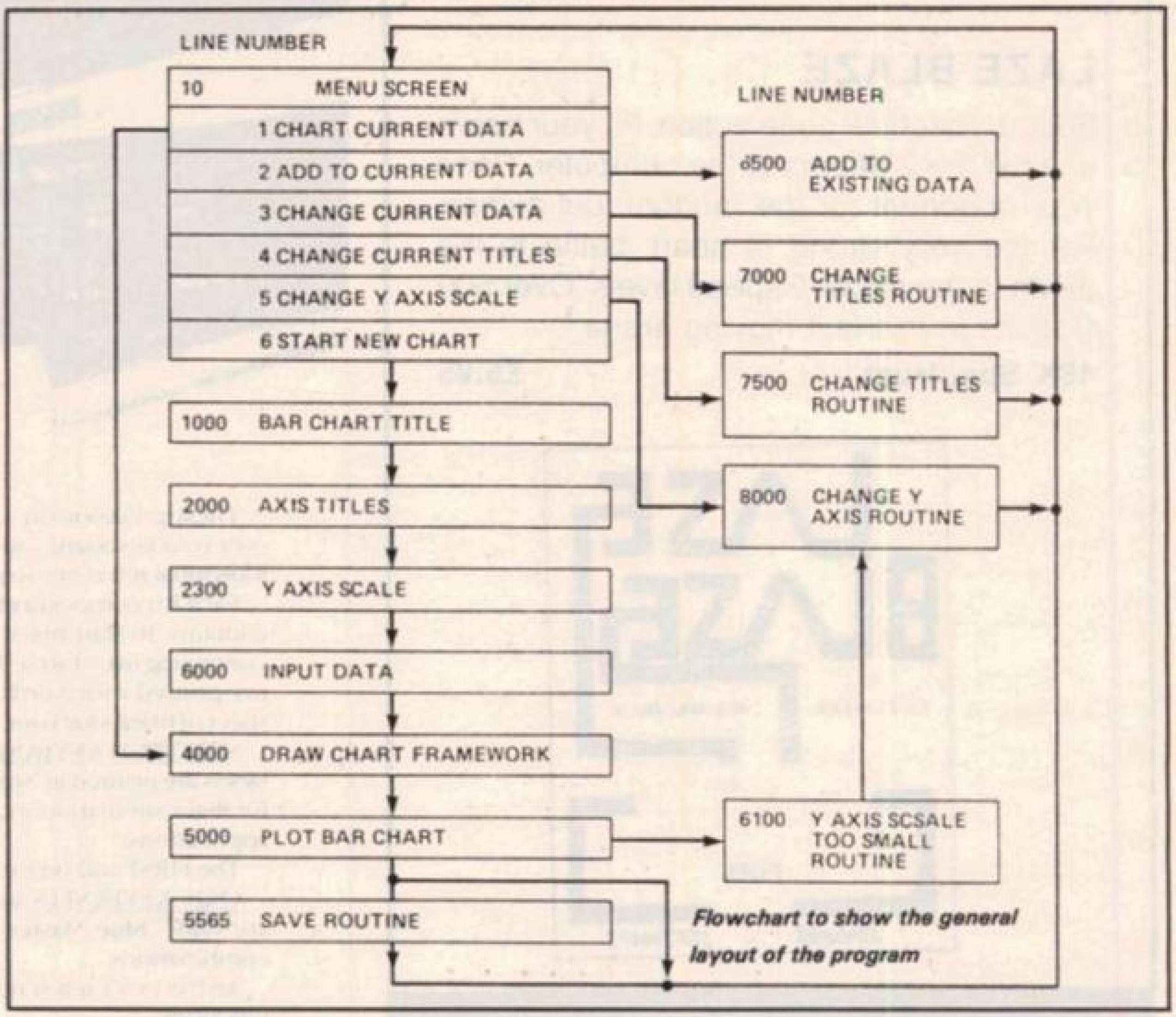
Compare electricity/gas usage quarter by quarter.

Chart bills such as telephone, rates, groceries etc.

Log petrol consumption, car mileage, service charges etc.

Present experimental results for school projects.

The list is really endless and I'm sure you can think of many more applications which would benefit from this method of presentation.



Flowchart to show the general layout of the program

Variables

V	Number of menu option selected.
T$	Bar chart title.
X$	X axis title.
W$	Column label.
Y$	Y axis title.
A	Maximum Y value.
B	Minimum Y value.
Z1	A-B modified to suit screen size.
Z2	
P	Power of 10 to which A-B has been modified.
Z(10)	String of scale numbers for printing on Y axis.
D$ to I$	Graphic variables for drawing chart framework.
C	Number of columns required.
D(C)	String of inputted data.
N(C)	String of inputted data less minimum value of Y (ie B).
C1	Number of extra columns required.
C2	Number of column where data is to be changed.

Loading instructions

Type in listing using line 10 REM "BAR CHART" to periodically SAVE as you go. Once this is completed, line 10 is redundant as the program offers an automatic SAVE option at line 5600. You might wish to change this to read 5600 SAVE T$ so that you SAVE and LOAD to the specific name you give to your chart.

Start the program by typing RUN 1000. This will then take you through every stage necessary to set up a new chart. (Alternatively type GOTO 10 which will start you at the menu screen and then choose option 6). From then on *never* RUN the program as this will destroy your variables. If for any reason you break out of the program then re-enter by typing GOTO 10.

You can always start a new chart on a different topic by LOADing an existing chart and selecting menu option 6.

A few words

1 INPUTted numeric data is rounded to the nearest pixel division on the Y axis scale.
2 By PLOTting the bars I amusing a grid of 20 rows by 24 columns.
3 The Y axis scale will never show a number greater than 100 but at the base of the scale you will see a scale factor eg X1, X10, X100 etc. Thus, a value of say 600 would show a bar to a height of 60 with a scale factor of X10.
4 I indicated in the opening remarks that the maximum number the program can handle is 9999. The graph can be used to plot *any* positive number but as the data is PRINTed above the graph I am limited in space to 4 digits per number to prevent adjacent numbers merging. You can INPUT non whole numbers providing the maximum number of digits does not exceed 4.

ie Data less than 1 =
numbers to 3 decimal places.

From 1 to 9.99 =
numbers to 2 decimal places.

From 10 to 99.9 =
numbers to 1 decimal places.

100 and above =
whole numbers only.

I have found this adequate for most applications.
5 As all INPUTted information is shown on the resultant bar chart (option 1 from menu) a hard copy of the graph is offered using the COPY feature at line 5730.
6 Using the "Electricity Consumption" shown in *Fig 1* you will note that I have INPUTted data as zero in columns 5, 10, 15 and 20 to delineate one year from the next and have labelled these columns with a SPACE. I have found this a useful presentation technique.

```
  10 REM "BAR CHART"
  11 REM * ALL CHARACTERS IN    *
         * LOWER CASE ARE TYPED*
         * IN INVERSE PRINT     *
  12 LET C1=0
  15 CLS
  18 PRINT TAB INT ((32-LEN T$)/
2);T$
  20 PRINT AT 2,11;"menu screen"
  30 PRINT
  40 PRINT " OPTIONS AVAILABLE .
...?"
  50 PRINT AT 7,4;"1. CHART CURR
ENT DATA"
  60 PRINT AT 9,4;"2. ADD TO CUR
RENT DATA"
  70 PRINT AT 11,4;"3. CHANGE CU
RRENT DATA"
  75 PRINT AT 13,4;"4. CHANGE CU
RRENT TITLES"
  80 PRINT AT 15,4;"5. CHANGE ""
Y"" AXIS SCALE"
  90 PRINT AT 17,4;"6. START A B
RAND NEW CHART"
  95 PRINT AT 21,0;" ENTER NO. O
F OPTION REQUIRED"
 100 INPUT V
 110 IF V=1 THEN GOTO 4000
 120 IF V=2 THEN GOTO 6500
 130 IF V=3 THEN GOTO 7000
 140 IF V=4 THEN GOTO 7500
 150 IF V=5 THEN GOTO 8000
 160 IF V=6 THEN RUN 1000
 170 IF V>6 OR V<1 THEN GOTO 100
 1000 REM *BAR CHART TITLE*
 1005 CLS
 1008 LET C1=0
 1010 PRINT "INPUT TITLE (MAX 32
 CHARACTERS)"
 1020 INPUT T$
 1027 IF LEN T$>32 THEN GOTO 1020
 1030 PRINT T$
 2000 REM *AXIS TITLES*
 2010 PRINT "INPUT X AXIS TITLE (
MAX 20 CHRS)"
2020 INPUT X$
2027 IF LEN X$>20 THEN GOTO 2020
2030 PRINT X$
2150 PRINT "LABEL COLUMNS (ONE C
HAR.PER COL)"
2152 PRINT AT 20,1;"123456789111
111111122222"
2154 PRINT AT 21,1;"          012
3456789012 34"
2160 INPUT W$
2162 IF LEN W$>24 THEN PRINT AT
19,0;"INPUT TOO LONG"
2164 IF LEN W$>24 THEN GOTO 2160
2165 PRINT AT 19,0;"
  "
2166 PRINT AT 20,1;"
           "
2168 PRINT AT 21,1;"
           "
2170 PRINT AT 8,0;W$
2250 PRINT "INPUT Y AXIS TITLE (
MAX 10 CHRS)"
2260 INPUT Y$
2267 IF LEN Y$>10 THEN GOTO 2260
2270 PRINT Y$
2300 REM *Y AXIS SCALE*
2310 PRINT "WHAT IS MAX Y VALUE?
 (NOTE:INPUT A POSITIVE WHOLE NO
. ONLY)"
2315 INPUT A
2316 IF A<1 THEN GOTO 2315
2318 PRINT A
2320 PRINT "WHAT IS MIN Y VALUE?
 (NOTE:INPUT A POSITIVE WHOLE NO
. ONLY....OR ZERO)"
2322 INPUT B
2324 IF B>=A THEN GOTO 2322
2326 PRINT B
2328 PRINT "PLEASE WAIT......"
2330 LET P=0
2350 LET Z1=(A-B)/10**P
2370 IF Z1>10 THEN LET P=P+1
2380 IF Z1>10 THEN GOTO 2350
```

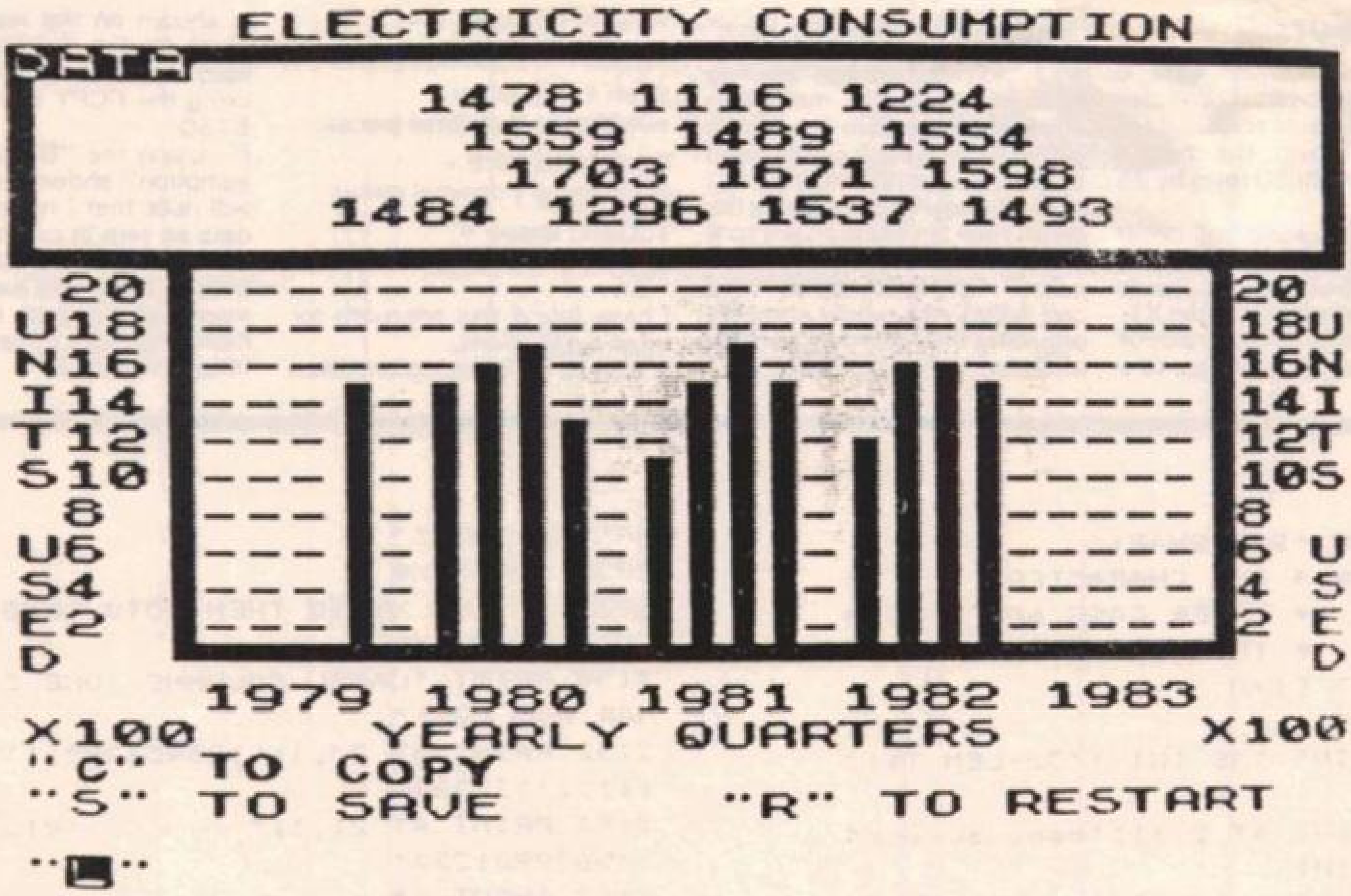


Fig 1.

```
2400 LET Z2=INT ((Z1+.05)*10)
2500 DIM Z(10)
2510 FOR R=1 TO 10
2520 LET Z(R)=INT ((Z2*R/10)+.5)
+INT (((B/10**P)+.05)*10)
2540 NEXT R
2600 GOSUB 6000
4000 REM *DISPLAY CHART*
4001 LET D$=""
4002 LET E$=""
4003 FOR I=1 TO 30
4004 LET D$=D$+CHR$ 3
4005 LET E$=E$+CHR$ 131
4007 NEXT I
4010 LET E$=CHR$ 130+E$+CHR$ 129
4015 LET H$=CHR$ 2+D$( TO 25)
4020 LET F$=CHR$ 133
4030 LET G$=CHR$ 5
4040 LET D$="data"+D$( TO 27)+CH
R$ 132
4050 LET I$="-------------------
-----"
4100 CLS
4110 PRINT TAB (32-LEN T$)/2;T$
4120 PRINT D$
4130 FOR R=1 TO 4
4140 PRINT G$;TAB 31;F$
4150 NEXT R
4160 PRINT E$
4200 FOR R=1 TO 10
4210 PRINT TAB 3;F$;TAB 4;I$;TAB
 28;F$
4220 NEXT R
4230 PRINT TAB 3;H$
4250 PRINT TAB 4;W$
4270 PRINT TAB 6+INT ((20-LEN X$
)/2);X$
4280 FOR R=1 TO LEN Y$
4290 PRINT AT 7+(INT (10-LEN Y$)
/2)+R,0;Y$(R)
4295 PRINT AT 7+(INT (10-LEN Y$)
/2)+R,31;Y$(R)
4300 NEXT R
4400 FOR R=1 TO 10
4410 IF Z(10)<=100 THEN PRINT AT
 17-R,1;Z(R);AT 17-R,29;Z(R)
4411 IF Z(10)>100 THEN PRINT AT
17-R,1;Z(R)/10;AT 17-R,29;Z(R)/1
0
4412 IF R-1=0 THEN GOTO 4420
4415 IF Z(R)=Z(R-1) THEN PRINT A
T 17-R,1;"  ";AT 17-R,29;"  "
4420 NEXT R
4430 IF Z(10)<=100 THEN PRINT AT
 19,0;"X";10**(P-1);AT 19,28;"X"
;10**(P-1)
4440 IF Z(10)>100 THEN PRINT AT
19,0;"X";10**P;AT 19,28;"X";10**
P
5000 REM *PLOT BAR CHART*
5010 FOR X=1 TO C
5015 IF D(X)<>0 THEN PRINT AT 1+
X-5*INT ((X-1)/5),3+X;D(X)
5020 FOR Y=1 TO INT ((N(X)*20/((
```

```
0**(P-1)*22))+.5)
5025 IF Y>21 THEN GOTO 6100
5030 PLOT 7+2*X,8+Y
5100 NEXT Y
5120 NEXT X
5500 PRINT AT 21,0;"""S"" TO SAV
E";TAB 16;"""R"" TO RESTART"
5510 PRINT AT 20,0;"""C"" TO COP
Y"
5550 INPUT K$
5560 IF K$="R" THEN GOTO 10
5561 IF K$="C" THEN GOTO 5700
5562 IF K$<>"S" THEN GOTO 5550
5565 PRINT AT 20,0;"1. SET TAPE
M/C TO RECORD MODE"
5570 PRINT "2. START TAPE M/C AN
D PRESS ""S"""
5580 IF INKEY$<>"S" THEN GOTO 55
80
5600 SAVE "BAR CHART"
5610 GOTO 10
5700 PRINT AT 21,0;"CHECK PRINTE
R AND THEN PRESS ""C"""
5710 PRINT AT 20,0;"                                "
5720 IF INKEY$<>"C" THEN GOTO 57
20
5725 PRINT AT 21,0;"
                                "
5730 COPY
5740 GOTO 10
6000 REM *INPUT DATA*
6005 DIM D(24)
6008 DIM N(24)
6010 CLS
6012 PRINT AT 4,0;"HOW MANY COLU
MNS? (MAX 24)       NOTE:INCLUDE
SPACES AS COLUMNS"
6014 INPUT C
6016 IF C>24 THEN GOTO 6014
6018 PRINT C
6020 PRINT AT 19,6;"input your d
ata now"
6025 FOR R=1 TO C
6030 SCROLL
6040 PRINT "DATA FOR COLUMN ";R;
" =   ";
6050 INPUT D(R)
6060 PRINT D(R)
6065 LET N(R)=D(R)-B
6070 NEXT R
6080 PAUSE 100
6090 RETURN
6100 REM *Y AXIS SCALE TOO SMALL
 ROUTINE*
6110 PRINT AT 21,0;"SCALE TOO SM
ALL......PLEASE WAIT"
6120 PAUSE 500
6130 GOTO 8000
6500 REM *ADD TO DATA ROUTINE*
6510 CLS
6515 IF C<>24 THEN GOTO 6600
6520 PRINT "DATA BASE IS FULL (2
4 COLUMNS)   ";"PUSH ""R"" TO RET
URN TO MENU PAGE."
6530 IF INKEY$<>"R" THEN GOTO 65
30
6540 GOTO 10
6600 PRINT AT 4,0;"HOW MANY EXTR
A COLUMNS?(MAX ";24-C;")    NOTE:
INCLUDE SPACES AS COLUMNS"
6630 INPUT C1
6640 IF C1>24-C THEN GOTO 6630
6650 PRINT C1
6720 PRINT AT 19,6;"input your d
ata now"
6725 FOR R=1 TO C1
6730 SCROLL
6740 PRINT "DATA FOR COLUMN ";C+
R;" =   ";
6750 INPUT D(C+R)
6760 PRINT D(C+R)
6765 LET N(C+R)=D(C+R)-B
6770 NEXT R
6775 LET C=C+C1
6780 PAUSE 100
6790 GOTO 10
7000 REM *CHANGE DATA ROUTINE*
7010 CLS
7020 PRINT "CURRENT DATA IS..."
7025 PRINT
7030 FOR R=1 TO INT ((C+1)/2)
7040 PRINT "COL.";R;"=";D(R)
7060 NEXT R
7070 FOR R=INT ((C+3)/2) TO C
7080 PRINT AT 1+R-INT ((C+1)/2),
16;"COL.";R;"=";D(R)
7090 NEXT R
7100 PRINT AT 15,0;"ENTER NO. OF
 COL. TO BE CHANGED"
7102 FOR R=1 TO 6
7104 PRINT AT 15+R,0;"
                                "
7106 NEXT R
7110 INPUT C2
7115 IF C2>C THEN GOTO 7110
7118 PRINT AT 16,0;C2
7120 PRINT AT 17,6;"input your d
ata now"
7130 PRINT
7150 PRINT "DATA FOR COLUMN ";C2
;" =   ";
7160 INPUT D(C2)
7170 PRINT D(C2)
7171 LET N(C2)=D(C2)-B
7172 IF C2<=INT (C+1)/2 THEN PRI
NT AT 1+C2,0;"               "
7175 IF C2<=INT (C+1)/2 THEN PRI
NT AT 1+C2,0;"COL.";C2;"=";D(C2)
```

```
7176 IF C2>INT (C+1)/2 THEN PRIN
T AT 1+C2-INT (C+1)/2,16;"
     "
7178 IF C2>INT (C+1)/2 THEN PRIN
T AT 1+C2-INT (C+1)/2,16;"COL.";
C2;"=";D(C2)
7180 PRINT AT 21,0;"ANY MORE CHA
NGES? (Y/N)"
7190 INPUT L$
7200 IF L$="Y" THEN GOTO 7100
7210 IF L$="N" THEN GOTO 10
7220 IF L$<>"N" THEN GOTO 7190
7500 REM *CHANGE TITLES ROUTINE*
7505 CLS
7510 PRINT "input title (MAX 32
CHARACTERS)"
7512 PRINT "CURRENT TITLE IS....
..."
7514 PRINT """";T$;""""
7515 PRINT "DO YOU WANT TO CHANG
E IT? (Y/N)"
7516 INPUT M$
7518 IF M$="N" THEN GOTO 7600
7519 PRINT "ENTER NEW TITLE...."
7520 INPUT T$
7527 IF LEN T$>32 THEN GOTO 7520
7530 PRINT T$
7600 PAUSE 100
7605 CLS
7610 PRINT "input x axis title (
MAX 20 CHRS)"
7612 PRINT "CURRENT TITLE IS....
..."
7614 PRINT """";X$;""""
7615 PRINT "DO YOU WANT TO CHANG
E IT? (Y/N)"
7616 INPUT M$
7618 IF M$="N" THEN GOTO 7634
7619 PRINT "ENTER NEW X AXIS TIT
LE...."
7620 INPUT X$
7627 IF LEN X$>20 THEN GOTO 7620
7630 PRINT X$
7634 PAUSE 100
7636 CLS
7640 PRINT "label columns (ONE C
HAR.PER COL)"
7644 PRINT "CURRENT COL. LABELS
ARE......"
7646 PRINT """";W$;""""
7650 PRINT "DO YOU WANT TO CHANG
E THEM?(Y/N)"
7660 INPUT M$
7670 IF M$="N" THEN GOTO 7780
7700 PRINT "ENTER NEW LABELS....
..."
7752 PRINT AT 20,1;"123456789111
1111111122222"
7754 PRINT AT 21,1;"         012
345678901234"
7760 INPUT W$
7762 IF LEN W$>24 THEN PRINT AT
19,0;"INPUT TOO LONG"
7764 IF LEN W$>24 THEN GOTO 7760
7765 PRINT AT 19,0;"
 "
7766 PRINT AT 20,1;"
          "
7768 PRINT AT 21,1;"
          "
7770 PRINT AT 16,0;W$
7780 PAUSE 100
7790 CLS
7850 PRINT "input Y axis title (
MAX 10 CHRS)"
7852 PRINT "CURRENT Y AXIS TITLE
 IS......"
7853 PRINT """";Y$;""""
7854 PRINT "DO YOU WANT TO CHANG
E IT? (Y/N)"
7855 INPUT M$
7856 IF M$="N" THEN GOTO 7900
7858 PRINT "ENTER NEW Y AXIS TIT
LE...."
7860 INPUT Y$
7867 IF LEN Y$>10 THEN GOTO 7860
7870 PRINT Y$
7900 GOTO 10
8000 REM *CHANGE Y AXIS ROUTINE
8005 CLS
8010 PRINT "WHAT IS MAX Y VALUE?
 (NOTE:INPUT A POSITIVE WHOLE NO
. ONLY)"
8015 INPUT A
8016 IF A<1 THEN GOTO 8015
8018 PRINT A
8020 PRINT "WHAT IS MIN Y VALUE?
 (NOTE:INPUT A POSITIVE WHOLE NO
. ONLY....OR ZERO)"
8022 INPUT B
8024 IF B>=A OR B<0 THEN GOTO 80
22
8026 PRINT B
8028 PRINT "PLEASE WAIT......."
8030 LET P=0
8050 LET Z1=(A-B)/10**P
8070 IF Z1>10 THEN LET P=P+1
8080 IF Z1>10 THEN GOTO 8050
8100 LET Z2=INT ((Z1+.05)*10)
8200 DIM Z(10)
8210 FOR R=1 TO 10
8220 LET Z(R)=INT ((Z2*R/10)+.5)
+INT (((B/10**P)+.05)*10)
8240 NEXT R
8250 FOR R=1 TO C
8260 LET N(R)=D(R)-B
8270 NEXT R
8300 GOTO 10
```

ZX COMPUTING

Lineage: 40p per word.

Semi display: £9.00 per single column centimetre. Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01 - 437 0699

Send your requirements to:
JANE EDMUNDS
ASP Ltd., 1 Golden Square,
London W1

SOFTWARE GAMES

~~**50 games for the Sinclair Spectrum, 16/48K. There are fifty super high quality games on one tape!!**~~

STOP PRESS:

There are now sixty games on this fantastic compilation, yes 60 super high resolution arcade games that you will have hours and hours of enjoyment from, guaranteed!!!

60 of the best games around including: Pacman, Invaders, Frogger, Fruit Machine, Defender, Lunar Lander, Pontoon, plus much more!! All on one tape for only £9.95.

New for the 48K Spectrum, "Flipwind". An amazin g simulation of the sidewinder arcade fruit machine, features include random holds and nudges, flipwinding reels to a winning line. Visually stunning graphics and 'state of the art' machine code, true arcade sound effects. Introductory price £5.50.

Still available 50 arcade games ZX81 £6.99

ONLY AVAILABLE BY MAIL ORDER FROM:

ANCO SOFTWARE (DEPT ZX)
25 Corsewall Street, Coatbridge,
Scotland ML5 1PX
Tel: 0236 32381
ANCO SOFTWARE . . . THE ULTIMATE EXPERIENCE

SPECTRUM GAME

Join the Spectrum Software Club now, and choose a **FREE GAME CASSETTE** for you to keep from our Top Twenty Best Sellers listed below. **OVER 150** tried-and-tested quality games available at a discount from all the top software houses — Only the **VERY BEST** programs are stocked. Our unique Exchange Plan means you can enjoy any game from our catalogue for up to 30 days for just **£1.00**. **EXCHANGE YOUR EXISTING GAMES** or the Free Game you get from us.

• NEWSLETTER • HARDWARE DISCOUNTS • REVIEWS ETC

*** TOP 20 TAPES ***

1	3D ANT ATTACK	48	11	3D SPACE WARS	16/48
2	FLIGHT SIMULATOR	16/48	12	TRANSAM	48
3	LUNAR JETMAN	48	13	MASTERCHESS	48
4	ARCADIA	16/48	14	FROGGER	16/48
5	KONG	16/48	15	AH DIDDUMS	16/48
6	PENETRATOR	48	16	MAZE CHASE	16/48
7	ZZOOM	48	17	BLASTERMIND	48
8	POOL	16/48	18	GOLF	48
9	QUEST ADVENTURE	48	19	MAZE CHASE	16/48
10	GULPMAN	16/48	20	3D ESCAPE	48

MEMBERSHIP APPLICATION

Please enrol me as a member of the Spectrum Software Club. I enclose £9.95 (payable to Video International) Send me the following Free Cassette.

Name________________

Address________________

________________P/code_____

Spectrum Software Club (ZX), Video International, 287-291 High Street, Epping, Essex CM16 4DA

NOVA SOFTWARE
presents
THE LAST SHIP OF ZENDOK
(48 ZX SPECTRUM)
£4.95

The first part of a 3 part adventure: **The Empire of Thailax**. Are you brave enough to challenge the evil Thailaxians? In this first text adventure, you must re-activate the last ship of the once great fleet of Zendok by finding 6 controlling crystals.

Please make cheques/POs payable to: **A. B. Saxton;** and send to: **Nova Software, (Dept ZX), 2 Gertrude Road, Draycott, Derby DE7 2NX.**

20% Discount
SPECTRUM 16K/48K AND
ZX81 Software
CONTACT
JAYCEE SOFTWARE
FREEPOST (NO STAMP)
FORRES IV36 0BR

SPECTRUM SOFTWARE

Large range of titles credit card orders by phone or ring for further details and price list.

FABULOUS PHONES,
72 High Street, Orpington,
Kent BR6 0VQ
Tel: (0689) 74051

TRADEWIND
48K SPECTRUM

COLOUR GRAPHIC ADVENTURE

Make a fortune selling and trading in the coral islands. Risk your life among pirates and sharks! Avoid reefs, weather storms and outwit the local traders.

£5 on OUR cassette or £4 on YOUR microdrive cartridges

WD SOFTWARE (Z), HILLTOP,
ST MARY, JERSY, C.I.
Tel: 0534 81392

FROM OLD KENT ROAD TO MAYFAIR

Play against your computer. "This simulator has been well thought out and works well." "Good value for money" - ZX Computing Oct/Nov 83. Only **£3.00** from: **M.P.T., 42 Raedwall Drive, Bury St Edmunds, Suffolk**

MAGIC SOFTWARE presents Tuck Shop a business game for any Spectrum £3.90. 27 Westwood Avenue, Dept C, Timperly, Cheshire. SAE.

MULTI INTERNATIONAL CO. REQUIRE SPECTRUM PROGRAMS

Our client, one of the most well known international companies require new, original programs for most home computers especially the ZX Spectrum. An excellent opportunity to obtain worldwide recognition.

Please contact:
Mr. R. J. Purssglove,
Marketing Consultants,
172 Finney Lane,
Healdgreen, Cheedle,
Cheshire
Tel: 061-437 0538

Replies treated in strictest confidence

THIS SPACE COULD BE WORKING FOR YOU

FOR SPECTRUM 48K:

WANT TO MAKE A MILLION?

Then play BOSSMAN: the exciting life game — involving, amongst others, production, wages, unions and natural disasters — where you make all the decisions **£5.95**

PIT-STOP: Become an expert racing driver picking up tools as you go but watch out for the oncoming computer cars — they're out to get you. **£4.95**

OMADAN ATTACK: Save Planet Mardon. Shoot through force fields with your lasers and astro-cannon, but beware — the Omandans fight back. **£4.95**

All games with Hi-Res colour graphics and full sound effects

Cheques/POs to:
AVALON SOFTWARE LTD., 62 Albion Street, Mansfield, Notts NG19 7AU

SPECTRUM KOPYKAT

IMPROVED PERFORMANCE

Simply the best. Copying **any** program is now as easy as LOADing and SAVEing your own programs. **Any ZX Spectrum** program can be backed-up onto a fresh tape. Even programs that cannot be stopped can now be copied. 100 per cent machine code. Prints file names. Copies headerless files. SPECTRUM KOPYKAT uses no program area so **full size** programs **over** 41.7K (9K for 16K machines) can be copied.
PLUS FREE Header Reader program **Despatched by first class return of post.** Send cheque/PO for **£4.95** to:
MEDSOFT, 61 Ardeen Road, Doncaster, South Yorks DN2 5ER. IT CAN EVEN COPY ITSELF.

TAPE COPIER 5

Makes back-up copies of all **Spectrum** programs (incl. headerless, m/c, unstoppable) that we are aware of with many unique features. **Full money back guarantee** if you are not satisfied.
* Loads in all program parts continuously. Verifies. Abort. Program names. Header data.
* Full instructions, very user friendly. **M/Drive Genuinely copies onto microdrive** – stops programs to help make them run.
Cost only **£4.50** or **£5.50** with M/Drive. (Overseas: +£1 Europe. £2 others).
5 Stars from the HCW Review. With 3 out of 4 ratings of 100%!
LERM, Dept ZX, 10 Grunswick, Gardens, Corby, Northants.

KEEP THE BANK MANAGER AT BAY
WITH . . . HOMAC ZX Home Accounting System

- Database for up to 1,000 records Spectrum: 3,000 for book-keeping and financial forecasting.
- Fast data entry.
- Manages most accounts: bank credit card, cash, savings, standing orders, direct debits etc.
- Predicts credit card bills.
- Up to 94 account headings covering an eighteen month period (Spectrum: 30 months) with annual carry forward.
- Printed or displayed summaries and statements for any account or period.
- Spectrum version provides for up to twenty currencies or individuals: ZX81 for any single currency only.

HOMAC £8.95

Please state if ZX81 (16K) or Spectrum (48K) version is required. Price includes cassettes and instructions. Post and packing **£1.05.**
DAVID MARTIN
Associates Ltd, 71A Princes Road, Richmond, Surrey TW10 6DQ
Telephone: 01-948 1053

HORSE RACE FORECASTING

This super programme picked the winners of the 1983 Grand National, St Leger and many more as first choice selections.

Spectrum 48K	**£12**
Spectrum 16K	**£10**
ZX 81 16K	**£10**
ZX 81 1K	**£8**

FOOTBALL RESULTS FORECASTING

Another great programme from Butronics. It forecasts home or away wins and draws, rating each one against the others.

Spectrum 48K	**£12**
ZX 81 16K	**£12**

Send your cheque or PO to: **The Butronics Co., Penjerrick House, Budock, Nr. Falmouth, Cornwall TR11 5ED**

VICTAGRAPH PLOT WINDOW

for ZX Spectrum & Tandy CGP 115 Printer Plotter (and interface)
TOTAL PRICE £8

New package and instructions: absolute system only. Position your design under the calibrated window, then place the window mask sight over each co-ordinate and read off X, Y for entry into Data. For display & plotter. **Victagraphics (ZX), 6A Bow Street, Rugeley, Staffs. U.K. WS15 2BT. Tel: (08894) 2426**

M&J SOFTWARE

SPECTRUM fig-FORTH (cassette) £12
A cassette-based implementation of FORTH. Complete with macro-assembler and powerful editor we believe this package to be excellent value for money. Also included completely free is a copy of the fig-FORTH Installation Manual.

DIY FORTH for the SPECTRUM **£12**
A Z80 fig-FORTH assembly listing and Installation Manual – all you need to put FORTH onto your Spectrum.
The language is many times faster than Spectrum basic as well worth the effort involved.

ALL ABOUT FORTH by Hydon **£7.95**
An excellent reference book with cross references to fig-FORTH, the FORTH-79 standard and Starting FORTH. This book should be next to every FORTH programmer's computer. (270+ pages).
Cheques and POs to:- **M&J Software, 34 Grays Close, Scholar Green, Stoke-on-Trent, ST7 3LU. Tel: (0782) 517876.**

CHARTSVILLE UK
Sinclair ZX81 16K

Check your knowledge of 25 years of worldwide popular music from rock and roll to the present day with this series of entertaining menu driven quizzes, each with 3D graphics score chart.

Chartsville UK 1957-1962
Chartsville UK 1963-1968
Chartsville UK 1969-1973
Chartsville UK 1974-1978
Chartsville UK 1979-1982

State your selection and send £4.50 cheque/PO per cassette (overseas £5.50) to:
MONTEREY SOFTWARE, 10 West Grove, Doncaster, DN2 5NB, England

SPECTRUM SPEEDYLOAD: Halve your waiting time! This short program lets you save/load at 3000 baud. Easy to use. Cassette (48K), £3.95. Ness Micro Systems, 100 Drakies Avenue, Inverness.

ZX-81 Shack-Pack cassette from Peibif. Morse and Rtty receiving program morse: 8-60 wpm. Rtty: 40-1000 Baud FL. 25 = or £6 = incl. postage. Jerry Holthaus, Irisstraat 73, 4542 Ed Hoek, Netherlands. Phone: 01154-1591.

SPECTRUM SYNTHESISER (48K). Metronome, drum sounds, rolling chords and piano options. By professional musician. £5.00. Scott Massie, No. 1 Bungalow, Firlands, Burghfield Common Berks.

SPECTRUM Trade Secrets. Includes making programs, unlistable, unstoppable, invisible, how to reveal such, revealing hidden Basic in machine code progams plus hints/tips, plus microdrive supplement – £3. ZX guaranteed (office ZC), 29 Chadderton Drive, Unsworth, Bury, Lancs.

SEND YOUR ADS TO JANE EDMUNDS

POOLS PREDICTION

Made easy on 48K Spectrum. Outputs Best Draws, Homes and Aways, features analytical draw finder which allows program to learn from its own mistakes, however small. Comes with free permutation calculator program.

128 Dividends won by writer so far.

(This is the best and most scientific of any program I have seen – Mr R. A. Pyatt – Secretary, Orpington Computer Club)

HORSE RACE PREDICTOR — a very successful predictor for the flat racing season

Ask for British Pools at £8.95
Australian Pools at £8.95
Ask for Race Predictor at £19.95

ROMBEST, 9 Hawkestone Ave, Newport, Salop, TF10 7SE

AUTO-DATA VEHICLE DIAGNOSIS FAULT FINDER

Suitable for most makes of vehicle. Extremely comprehensive covering engine, gearbox (manual and automatic), brakes, suspension, steering and electrical system. Superb value at £5.50 inc. VAT. Cheques and POs to:

ERC COMPUTER SERVICES (Dundee), 53 Thurso Crescent, Dundee, Scotland. Tel: 0382-642017

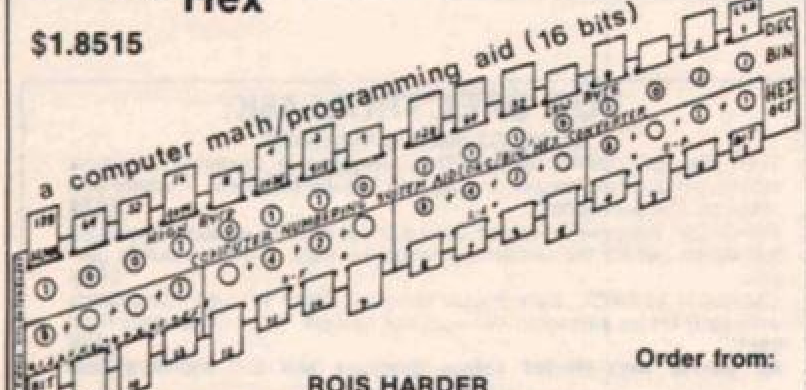
TARDIS COMPUTING

TIME LEDGER 48K Spectrum **£9.** For accountants & solicitors. Fully menu driven; 600 clients and 17 staff with expenses and client names. Saves hours!

Spectrum 16K
MICRODRIVE INDEX **£4.50.** Loads progs. from Microdrive.
TRANSPORT & SHAPES **£4.50.** Educational for children.
All progs. Microdrive compatable 4, **Preston Gardens, Ilford, Essex.**

DecBinHex Converter

$1.8515



Order from:
ROIS HARDER
995 Shakespeare Avenue,
N. Van., B.C., Canada V7K 1E7 (604) 980 4167

FOR ALL YOUR SOFTWARE NEEDS. RING 01-437 0699 X338

SOFTWARE EDUCATIONAL

LEARN ABOUT ELECTRONICS WITH YOUR SPECTRUM 48K

New structured independent learning package for beginners — designed by electronics engineers — includes:

- Computer generated animated graphics
- Learners guide
- Kit of components
- Practical guidance and exercises

Package suitable for beginners GCE/CSE, C&G, etc and covers basic semi-conductor theory. £9.95 incl. post and packing.

E.T.S.T., 117 Loughborough Road, Kirkcaldy KY1 3DD

Ed.SOFT Educational Software

76 Woodville Road, Exmouth EX8 1SW
Tel: (0395 275741)

Cassette based Educational Software for Spectrum micro. Suitable for all from 7 to 13. Programs from **£4.95** include: **Whatsit** (20 questions game), **Wordflash** (to aid the increase of reading age), **Mmaths/Placeval** (add, subtract, divide, multiply, check place values). **C. Rules** (£5.95 information, testing and reinforcement of rules for hard & soft 'c'). Also **Anagram** and **Basically Basic** Booklet (£1) & others.

POPPY PROGRAMS ZX Spectrum (16 or 48K)



HIDDEN LETTERS Age 5-adult **£6.00**
Highly educational reading game, 9 skill levels.
MONEY Age 3-10 **£6.50**
Coin recognition, 4 easy levels teach money sums
TELLING THE TIME Age 3-10 **£7.00**
6 programs for learning in easy stages, 12, 24 hour, digital time, time in words, clocks to set.
LEARNING TO READ 1 Age 2-6 **£6.50**
4 All graphics Pre-Reading games
All cassettes include special keyboard overlays. P&P inc.
Richmond House, Ingleton, via Carnforth, Lancs LA6 3AN

TINO COPY

For 48K or 16K Spectrum

Why be satisfied with just a tape copier?

Get our 3-in-1 utility featuring:-
COPY **ANY** CASSETTE PROGRAM
PLUS
DOUBLE SPEED LOAD/SAVE UTILITY
PLUS
CASSETTE 'HEALTH RATING' UTILITY

Copier does it all — MULIPLE sections, HEADER printing, MULTIPLE copies. Full SIZE programs. Double speed utility halves your load/saves time. 'Health Rating' program detects possible faults in your cassette recorder *before* they become serious. Also useful for finding defects in tape. Avoid wasted SAES & lots of frustration.

At £4.50 this must be the best value.
CAMEL MICROS, 33A Commercial Road, Weymouth, Dorset

ZX81 (16K) HIDDEN LETTERS

Word game helps reading ability.
£4.95 inc from
Poppy Programs

NEW DRIVING TEST (Part 1)

Simulated controls, faults analysis. Good graphics, a useful game if you are learning or interested in driving. ZX81 (16K) **£5.95.** 16K Spectrum **£6.95.** Electron **£7.50.**

Replies to: DEPT ZX
ACE SERVICES, 8 APERFIELD ROAD, BIGGIN HILL, KENT
Trade Enquiries Welcome

BEGINNING PIANO, or keen on text adventures? Send SAE to Fair Software, 30 Fair Lane, Robertsbridge, East Sussex.

KILSOFT

48K SPECTRUM: Educational revision programs, GCE 'O'/CSE Biology, Physics, Chemistry, Maths etc. £4.95 each. For FULLEST details: Kilsoft, 32 Briarlyn Avenue, Birchencliffe, Huddersfield. Tel: (0484) 31491

REPAIRS

ZX81 SPECTRUM. We can now offer out-of-guarantee repairs by our Computer Dept. Our engineers have had over three years experience servicing Sinclair Computer products. Price including p&p. ZX81 £11.50, 16K Ram £9.95, Spectrum £18.75. Send with cheque or PO. T.V. Service of Cambridge, French's Road, Cambridge CB4 3NP. Phone (0223) 311371.

COMPUCARE Sinclair repairs specialists all makes speedily repaired at reasonable cost. 20 Shirley Rd., Kings Norton, Birmingham. Phone 021-458 7188 (6-8pm) weekends.

LIBRARIES

SPECTRUM OWNERS

Why not use our software hire service. With an extensive range of games, educational programs, etc. Only 50p per tape plus P&P.
Send SAE for full details to:
X.E. Lent Software,
8 Cheviots, Highfield,
Hemel Hempstead HP25RD

OUR READERS NEED YOUR SERVICES RING 01-437 0699 X338

UTILITIES

SUPERIOR TAPE COPIER

We are convinced that the new 007 SPY is the very best Spectrum tape copier available on cassette. **Find a better one offering better value for money, and TWICE your money will be refunded.**
A. Can copy multipart programs in one go automatically.
B. Can copy genuine 48K long programs.
C. Genuinly the easiest copier to use.
D. Has copied all the programs we know of.
E. Microdrive compatible AND *includes program to transfer tapes* to the Microdrive AND they'll work.
F. Post free even to Europe.

007 SPY IS JUST £2.95

ZX GUARANTEED (Dept ZX), 29 Chadderton Drive, Unsworth, Bury, Lancs
Tel: 061-766 5712

(S)TOP SECRET(S) for any Spectrum

Be in full control of programs on your computer. STOP and reveal the SECRETS of your favourite programs. A machine code UTILITY that STOPS, most Basic or Machine Code programs, protected or not, WITHOUT CRASHES.
Activate routine, press two keys and be in FULL CONTROL/LEARN/LIST/MODIFY etc., all the options are yours. The ideal MICRODRIVE companion.
If you're a basic beginner, a machine code master, a program writer or a games fanatic, this is the UTILITY that can open the door for YOU.
PLUS: pages of SECRET information about methods used by professionals to keep their secrets hidden.
48K SPECTRUM **£5.95**, including HEADER READER and Dec/Hex Hex/Dec converter.
16K SPECTRUM **£4.95** (Relocatable under software control).
Cheque/PO to:
ASH BYPRODUCTS, Dept ZX, 183 Bristol Road, Edgbaston, Birmingham B5 7UB

G. A. BOBKER Does it again

(How long before the competitors catch-up with this super program)? New TAPE 3 lets you STOP Headerless Files and unmergable Basic programs so you can study them and/or put to Microdrive. Includes a pseudo disassembler which clearly gives all the relevant info in plain English.

TAPE 3 £3.50 (£3 to club members)

ZX GUARANTEED (Dept ZX)
29 Chadderton Drive, Unsworth, Bury, Lancs. Tel: 061-766 5712

COURSES

NEW TO COMPUTING

We are here to help you gain skills in computing. Individual machines, qualified tuition, practical course. Ring Computer Workshop on 01-318 5488. 4 Lee High Road, Lewisham, London SE13 5LQ
Childrens' Holiday Courses also available.

SERVICES

ATTENTION

all 16K Spectrum owners. Have your Spectrum internally expanded to 48K. Does not affect warranty. All work guaranteed for **£35** including VAT, fitting and return postage. 24 hour turnaround.
Send cheque/postal order and Spectrum alone (no wires power pack necessary) to:
Barnsley Micro Computers, (Dept ZX), 196 Carlton Rd., Smithers, Barnsley, South Yorkshire or phone **0226 4637 for further details.**

CLUBS & USER GROUPS

MICRODRIVE USERS CLUB

Called **Microdrive Exchange** this newsletter club is designed to help by the exchange of info and ideas. Did you know you can pay up to £5.50 for a cassette to transfer tapes to microdrives, and then they don't run. Issue 1 will include how to avoid this plus transferring **Vu-File**, **ZZOOM** etc. Membership is £5 per year. Europe £8 per year.
Please make cheques payable to ZX Guaranteed. **Microdrive Exchange** is run by C. A. Bobker.

ZX GUARANTEED (Dept ZC)
29 Chadderton Drive, Unsworth, Bury, Lancs.

SWAP SHOP AND LIBRARY

Hundreds of titles available for exchange or hire as well as other services. For details send large SAE to:
ZX Section, Personal Software Users' Assoc., 19 Lammerton Terrace, Dundee DD4 7BR Scotland

---

PHONE JANE EDMUNDS TO BOOK SPACE ON 01-437 0699 X338

NATIONWIDE SHOPS & DEALERS

LANCASHIRE MICROS

89 Euston Rd, Morecambe, Lancs
TEL: 411435. Also open Sundays.

Sinclair, Commodore, Acorn/BBC + Dragon + Lynx.
Authorised dealer for Bug-Byte, Imagine, Quicksilva, Artic, Melbourne House, Silversoft etc.

ACCESSORIES

CLASSIFIED ADVERTISEMENT — ORDER FORM

1.	2.	3.
4.	5.	6.
7.	8.	9.
10.	11.	12.
13.	14.	15.

Please place my advert in ZX Computing for .... months
Please indicate number of insertions required.

ONLY 40p per word (minimum charge 15 words)

CLASSIFIED DEPT., ZX COMPUTING,
1 Golden Square, London W1.
Tel: 01-437 0699

Name ....

Address ....

....

....

Tel. No. (Day) ....

ADVERTISERS INDEX


ASP SOFTWARE
Presents
The Greatest Challenge Ever To Face Man And Machine
(Atari, BBC B, Spectrum and Commodore 64)
FIVE MILLION BARBARIANS
AN ARMY OR TWO OF PERSIANS
SEVERAL THOUSAND UPSET ARMENIANS
FIFTY LEGIONS OF ITINERANT ITALIANS
..................AND YOU!
THE FALL OF ROME
SPQR
In case of difficulty, Order from: Argus Press Software Group, No 1 Golden Square London W1R 3AB
Please add 50p p&p and allow 14 days for delivery
At your dealers now!
£6.99 all inc.
HORNSBY 84

NOT JUST A PRETTY FACE!

Programmable Joystick Interface



This superb new interface is one of our very latest developments for your Spectrum. Offering even more features, and as it's programmable from the keyboard or with the cassette supplied you can now use it with any software.

Features include

- Eight directional movement
- Keyboard fully functional
- Rear connector for other add-ons
- Microdrive compatible



ONLY £22.95

Quickshot Joystick I®



- Super positive response
- Stabilizing suction caps
- 2 fire buttons
- 4ft lead

ONLY £9.99

Please rush me the following

.... Programmable joystick interface ..... @ £ ....

.... Quickshot Joystick I .... @ £ ....

.... Quickshot Joystick II .... @ £ ....

Please add post and packing £1.25

I enclose cheque/PO/Cash for ...... Total £ ....

or debit my Access/Barclaycard No.

Signature ....

Name ....

Address ....

Quickshot Joystick II®



Incorporating all the features of 'Quickshot I' plus

- Improved control grip
- Trigger fire button
- Rapid fire option

ONLY £12.95

dk'tronics

DK Tronics Ltd., Unit 6, Shire Hill Industrial Estate, Saffron Walden, Essex CB11 3AQ. Telephone: (0799) 26350 (24 hrs) 5 lines

The Spectrum Connection